INTEGRATING SYNTAX, SEMANTICS, AND DISCOURSE
DARPA NATURAL LANGUAGE UNDERSTANDING PROGRAM

R&D STATUS REPORT

SDC -- A BURROUGHS COMPANY

ARPA ORDER NUMBER: 5262
PROGRAM CODE NO. NR 049-602 dated 10 Aug 1st 1984 (433)
CONTRACTOR: System Development Corporation CONTRACT AMOUNT: $683,105
CONTRACT NO: N00014-85-C-0012
EFFECTIVE DATE OF CONTRACT: 4/29/85 EXPIRATION DATE OF CONTRACT: 4/29/87
PRINCIPAL INVESTIGATOR: Dr. Lynette Hirschman PHONE NO. (215) 648-7554

SHORT TITLE OF WORK: DARPA Natural Language Understanding Program

REPORTING PERIOD: 2/1/86 - 4/30/86

1. Description of Progress

1.1. Grammar

1.1.1. Intermediate Syntactic Representation

A Prolog version of the representation developed at NYU to mediate between the syntactic parse and semantics (the intermediate syntactic representation, or ISR) has been implemented, and ISR rules for the SDC restriction grammar have been developed. This provides a uniform syntactic output for the SDC and NYU systems.

1.1.2. BNF Rules and Restrictions

Coverage of CASREPS

Out of 154 total sentences in the CASREP corpus, 131, or 85%, are correctly parsed. 92 of these are parsed correctly on the first parse, 17 on the second or third parse, and 22 on the fourth or subsequent parse. 23 are not parsed correctly, either due to ill-formed input, problems with the lexical scanner (discussed below), inadequacies of grammar coverage, or xor problems (also discussed below).

Extensions to Grammar

The extensions to the grammar required to parse the CASREPS corpus include the addition of rules for fragments, objects, sentence adjuncts, and "wh-constructions" such as relative clauses. A detailed discussion of the grammar extensions and parsing results for the CASREP sentences is included in the appendices to this report.

Fragments

Approximately half of the sentences in the CASREPs are not full sentences. Nevertheless, these fragments follow quite regular patterns, and fall into one or another of four basic types: *tvo* (tensed sentence missing subject, as in A4.1.2, *Believe the coupling from diesel to sac lube oil pump to be sheared*); *zerocopula* (missing verb *be*, as in A6.0.0, *Part ordered*); *nstg_fragment* (isolated noun phrase, as in B34.1.1, *Loss of oil pump pressure*); or *predicate* (isolated complement of verb *be*, as in B12.1.2, *Believed due to worn bushings*, or A.1.1.2, *Unable to consistently start nr 1b gas turbine*).

The syntax and the semantics of these elements are quite regular, and thus fragment coverage does not add signficantly to the complexity of the grammar. A total of six BNF rules (out of 106 total) and 3 restrictions (out of 55 total) were added to the grammar to cover fragments; in addition, 2 BNF rules and 1 restriction were altered to accomodate fragments.

Object Options

The grammar has also been extended to cover a wider range of object types, including a variety of embedded infinitivals, embedded clauses, and non-clausal predications such as *subject+object of be* (as in B26.1.5, *High lo temp due to design of first flight oil cooler believed contributor to unit failure*).

Sentence Adjuncts

A rich variety of sentence adjuncts occurs in the CASREPS, including a range of clausal and sub-clausal strings introduced by subordinating conjunctions (as in B20.1.1, *while engaged*) and present participles (as in B11.1.1, *causing erratic operation*). In addition, the restriction component was extended to prevent spurious ambiguities arising out of the enrichment of sentence adjunct possibilities.

Wh-expressions

Relative clauses and other wh-expressions are rare in the CASREPs. However they do occur (cf. B36 1.3, *65 psi which is low lube oil alarm set point*);the grammar has also been expanded to cover these constructions and to enforce the complex restrictions on their occurence.

Problems

The major remaining difficulties include the following:

Lexical scanner problems

Word-internal occurences of periods, slashes, etc. are currently rejected by the lexical scanner.

XOR Problems

The 'committed or' which controls disjunctive application of the assertion, question, fragment, and compound options is generally successful in capturing the intended parse. However, there are several sentences in the CASREP corpus in which a spurious assertion parse preempts a correct fragment parse, e.g., B26.1.5, *High lo temp believed contributor to unit failure.*, where *believe* is taken as the main verb with subject *temp* and *contributor* as the object (*they believed it*), rather than as a fragment of the type *zero_copula*, where *believed* is taken as a past participle (*temp [was] believed [to be] a contributor...*).

Remaining grammar problems

Full and accurate coverage of the CASREPs requires further work on the grammar, including the following: finer-grained treatment of the noun phrase; restrictions on adverbs to prevent, e.g., the analysis of *very* as a sentence adverb; modification of the BNF rules to accomodate multiple sentence adjuncts; modification of conjunction rules.

1.2. Semantics

Semantic Coverage

Approximately 150 lexical items have been identified in the CASREPS corpus which need specialized semantics rules. These include verbs, nominalizations, and nouns with arguments, as discussed below. Rules have been developed for 83 of these lexical items, primarily those having to do with machine states and functions, about half of the total.

Interpreter Modifications

The processing of nominalizations and verbs is being made more and more distinct.

An unbound obligatory role now causes backtracking and reassignment of syntactic constituents.

An extra level has been added to the interpreter to allow for the recognition of transparent predicates, and for the call to the temporal component. These transparent predicates do not have decomposition rules but their arguments do. This makes it possible to represent components of meaning pertaining to the temporal properties of verbs (aspectual operators such as *start* and *occur*) and also to handle verbs whose complements provide the semantic content of a predication (*have* and *be*).

Extensions to Semantics

Nominalizations and Nouns with Arguments

The coverage of the domain specific semantics has been expanded to include nominalizations and nouns which take arguments. The verb semantics component has been generalized to handle several types of noun phrases whose semantics resembles that of sentences. Nominalizations, such as *clutch engagement* and *engine start*, can be analyzed, as well as nouns which take arguments, such as *oil pressure*. The final semantic description of a noun phrase such as *clutch engagement* resembles that of the related sentence, *clutch engages*. The syntactic differences between the sentence and noun phrase is captured by having two sets of mapping rules, the rules which relate syntactic constituents to semantic roles, one set for sentences and one set for noun phrases. In the case of the verb, *engage*, the mapping rules specify that the subject can be mapped to the patient role. In the case of the noun, *engagement*, the corresponding mapping rule specifies that a noun modifier can be mapped to the patient role. Nominalizations go through time analysis (discussed below) just as do sentences.

Certain nouns (e.g., *temperature* and *pressure*) have an argument structure similar to verbs and nominalizations. They have their own decomposition rules but make use of general noun phrase mapping rules. Thus noun phrases like *oil pressure in sac* can be handled somewhat analogously to nominalizations like *metal contamination in oil filter*. The processing for nouns with arguments differs from nominalizations in that nouns with arguments do not go through time analysis.

Verb Taxonomy

The verbs have been analyzed according to several criteria in order to assign them to categories in a verb taxonomy. The criteria include the semantic classes of the verb arguments, the semantic roles of the same arguments, the possible syntactic realizations of those semantic roles, and the semantic usage of the verb in this domain. During this process the set of semantic roles, the set of semantic classes for verb arguments, and the set of semantic categories for the verb taxonomy have been gradually stabilizing, and are discussed in more detail in the Appendix ?.

Semantics Rules

Decomposition rules and corresponding mapping rules for both noun phrases and clauses have been designed for several classes of verbs, nominalizations and nouns.These classes include investigative activities, measurements of pressure and temperature and changes of measure, maintenance activities, symptoms in or damage to machine parts or systems, and repair, removal or installation of machine parts. These rules are being gradually added to the working system in order to insure smooth interaction among the clause semantics, noun phrase semantics, reference resolution and the time component.

1.3. Pragmatics

Reference Resolution

A detailed discussion of cooperation between semantics and reference resolution is provided in the paper, *Recovering Implicit Information,* which is included as an appendix. The paper, *Focusing and Reference Resolution in PUNDIT,* which describes the reference resolution process in detail, is also included as an appendix.

Temporal Analysis Component

A domain independent component to process information about time has been implemented. The time component cooperates closely with the semantic analysis of noun phrases and clauses. Temporal information is present in the inherent meaning of the verb or nominalization, the tense of the main verb, the perfect or progressive verbal elements if present, and the meaning of time adverbs. The time component takes the output from the semantic analysis of the main clause of every sentence, and of references to events in adverbial phrases expressing a time relation, and processes the temporal information contained in the sentence. Because a clause or nominalization can refer to a real or hypothetical state-of-affairs, the time component must first determine whether or not a unique, specific time has been referred to. If so, it then determines the temporal properties of the real-time states-of-affairs and the temporal orderings of the various states-of-affairs.

While the past tense of verbs without modal auxiliaries generally refers to a specific time, the present tense has real-time reference only with certain verbs. For verbs in the present tense, the current implementation determines whether a real time has been referred to by looking at the meaning of the verb. Future implementations will also look at modal verbs (e.g. *will/should/can*) and the presence of adverbs which have inherent time reference (e.g., yesterday/May 18, 1986).

Three types of states-of-affairs with different temporal properties are represented: 1) states, 2) processes, and 3) changes-of-state. A state is a situation in which there is no change from moment to moment, i.e., the state remains constant through some PERIOD of time. A process is a state-of-affairs in which there is change from moment to moment, i.e., some kind of activity takes place over some PERIOD of time. A change-of-state predication denotes a transition at some MOMENT in time to a new state-of-affairs.

Every state-of-affairs takes place at some event time (ET). There are two temporal orderings that can be computed for every real-time state-of-affairs: the relation of event time (ET) to the time at which the text was generated (an obligatory relation) and the relation of its event time to the times of reference events (RT) mentioned in adverbial phrases or clauses (an optional relation). States and processes can precede, be contemporaneous with, or start before and continue through the time at which the report is generated (GT):

engine was operating	(ET precedes GT)
engine is operating	(ET contemporaneous with GT)
engine has been operating	(ET starts before and continues through GT)

The outcome of a change-of-state can itself be a state or a process and thus may also have the relations to report generation time given above.

Given a time adverb which relates two states-of-affairs, (e.g., *before/after/when*) the time component computes the relative ordering on the basis of the meaning of the adverb and the temporal properties of the relevant states-of-affairs. The set of possible relations between two states-of-affairs ET and RT currently includes:

sac disengaged immediately after alarm.	ET after RT
pressure dropped to 72 psi then increased to 90 psi.	RT after ET
drive shaft remained stationary while hub continued to rotate.	ET overlaps RT
the drive shaft was packed with 60 grams of grease when it was installed.	ET same as RT
failure occurred during engine start.	ET during RT
the diesel was operating when the alarm sounded.	RT during ET

1.4. Facilities

A window system is under development on the Symbolics for displaying the output from PUNDIT, the parse tree, and various trace messages. This will considerably enhance our development environment. In addition, it will provide a convenient format for presenting demonstrations of the system.

We have received Release 12.11 of Symbolics Prolog and it has been installed.

Prolog for the Government furnished Symbolics machine has arrived, and has been installed.

2. Change in Key Personnel

--none--

3. Summary of Substantive Information from Meetings and Conferences

3.1. Professional Meetings Attended

--none--

3.2. SDC/NYU Meeting

SDC/NYU Meeting #8 (April 4, New York University, New York, NY)

Lynette Hirschman, Martha Palmer, Rebecca Schiffman and Deborah Dahl went to New York to meet with Ralph Grishman, Tomasz Ksiezyk, Dimitri Turchin, Ngo Thanh Nhan, and Leo Joskowicz. Palmer gave a presentation on verb semantics. Schiffman gave a presentation on the analysis of time in the CASREPS. Leo Joskowicz discussed domain inference rules which he has developed for SAC malfunctions.

3.3. DARPA Meetings

Meeting of Strategic Computing Natural Language Contractors

A meeting of the natural language contractors was held May 1-2 at ISI. During the meeting, each of the seven contractors (BBN, ISI, NYU, SDC, SRI, U. Massachusetts and U. Pennsylvania) gave an hour presentation, and several of the contractors (BBN, ISI, Penn, NYU) also gave demos of the systems under development. In addition to the natural language contractors, there were also presentations from two Expert Systems contractors (Teknowledge and Ohio State) and from two Speech contractors (CMU and BBN). The overall focus was on exchange of technical information, but the meeting concluded with an afternoon session for the Natural Language Principal Investigators. At this smaller meeting, a number of issues were discussed, including status of the follow-on contracts, the need for

proposals for the follow-on contracts, contractors estimates of the impact of possible budget cuts, possibilities for interaction with the Air Land Battle Management program and a recommendation for an annual natural language meeting, including all DARPA natural language contractors, not just Strategic Computing.

3.4. Symbolics Lisp User's Group

John Dowding attended a meeting of the Mid-Atlantic Division of the Symbolics Lisp User's Group in April at the University of Pennsylvania. The meeting included a presentaion on Symbolics networking software.

4. Problems Encountered and/or Anticipated

Although the Symbolics Prolog development environment has improved, there are still problems with the development environment and debugging facilities.

5. Action Required by the Government

6. Fiscal Status

(1) Amount currently provided on contract:
$672,833 (funded) $683,105 (contract value)

(2) Expenditures and commitments to date:
$ 295,202

(3) Funds required to complete work:
$387,903

APPENDIX

Quarterly Progress Report #4

Integrating Syntax, Semantics, and Discourse

DARPA Natural Language Understanding Program

SDC -- A Burroughs Company

May 15, 1986

CONTENTS:

Focusing and Reference Resolution in PUNDIT

Deborah A. Dahl

Research and Development Division
SDC -- A Burroughs Company
PO Box 517
Paoli, PA 19301

To be presented at AAAI-86, August 11-15, 1986, Philadelphia, PA

ABSTRACT

This paper describes the use of focusing in the PUNDIT text processing system.[1] Focusing, as discussed by [Sidner1979] (as well as the closely related concept of centering, as discussed by [Grosz1983]), provides a powerful tool for pronoun resolution. However, its range of application is actually much more general, in that it can be used for several problems in reference resolution. Specifically, in the PUNDIT system, focusing is used for *one*-anaphora, elided noun phrases, and certain types of definite and indefinite noun phrases, in addition to its use for pronouns. Another important feature in the PUNDIT reference resolution system is that the focusing algorithm is based on syntactic constituents, rather than on thematic roles, as in Sidner's system. This feature is based on considerations arising from the extension of focusing to cover *one*-anaphora. These considerations make syntactic focusing a more accurate predictor of the interpretation of *one*-anaphoric noun phrases without decreasing the accuracy for definite pronouns.

[1] This work is supported in part by DARPA under contract N00014-85-C-0012, administered by the Office of Naval Research. APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED

1. Background

1.1. Focusing

Linguistically reduced forms, such as pronouns, are typically used in texts to refer to the entity or entities with which the text is most centrally concerned.[2] Thus, keeping track of this entity, (the ***topic***, of [Gundel1974], the ***focus*** of [Sidner1979], and the ***backward-looking center*** of [Grosz1983, Kameyama1985]) is clearly of value in the interpretation of pronouns. However, while 'pronoun resolution' is generally presented as a problem in computational linguistics to which focusing can provide an answer (See for example, the discussion in [Hirst1981]), it is useful to consider focusing as a problem in its own right. By looking at focusing from this perspective, it can be seen that its applications are more general than simply finding referents for pronouns. Focusing can in fact play a role in the interpretation of several different types of noun phrases. In support of this position, I will show how focus is used in the PUNDIT (Prolog UNDerstander of Integrated Text) text processing system to interpret a variety of forms of anaphoric reference; in particular, pronouns, elided noun phrases, ***one***-anaphora, and context-dependent full noun phrase references.

A second position advocated in this paper is that surface syntactic form can provide an accurate guide to determining what entities are in focus. Unlike previous focusing algorithms, such as that of [Sidner1979], which used thematic roles (for example, ***theme, agent, instrument*** as described in [Gruber1976]), the algorithm used in this system relies on surface syntactic structure to determine which entities are expected to be in focus. The extension of the focusing mechanism to handle ***one***-anaphora has provided the major motivation for the choice of syntactic focusing.

The focusing mechanism in this system consists of two parts--a **FocusList**, which is a list of entities in the order in which they are to be considered as foci, and a focusing algorithm, which orders the **FocusList**. The implementation is discussed in detail in Section 5.

1.2. Overview of the PUNDIT System

I will begin with a brief overview of the PUNDIT system, currently under development at SDC. PUNDIT is written in Quintus Prolog 1.5. It is designed to integrate syntax, semantics, and discourse knowledge in text processing for limited domains. The system is implemented as a set of distinct interacting components which communicate with each other in clearly specified and restricted ways.

The syntactic component, Restriction Grammar,[Hirschman1982, Hirschman1985], performs a top-down parse by interpreting a set of context-free BNF definitions and enforcing context-sensitive restrictions associated with the BNF definitions. The grammar is generally modelled after that developed by the NYU Linguistic String Project [Sager1981]. Restrictions which enforce context-sensitive constraints on the parse are associated with the bnf rules

---

[2] I am grateful for the helpful comments of Lynette Hirschman, Marcia Linebarger, Martha Palmer, and Rebecca Schiffman on this paper. John Dowding and Bonnie Webber also provided useful comments and suggestions on an earlier version.

Some semantic filtering of the parse is done at the noun phrase level. That is, after a noun phrase is parsed, it is passed to the noun phrase semantics component, which determines if there is an acceptable semantics associated with that parse. If the noun phrase is acceptable, the semantics component constructs a semantic representation. If the noun phrase is not semantically acceptable, another parse is sought.

At the conclusion of parsing, the sentence-level semantic interpreter is called. This interpreter is based on Palmer's Inference Driven Semantic Analysis system, [Palmer1985], which analyzes verbs into their component meanings and fills their thematic roles. In the process of filling a thematic role the semantic analyzer calls reference resolution for a specific syntactic constituent in order to find a referent to fill the role. Reference resolution instantiates the referent, and adds to the discourse representation any information inferred during reference resolution.

Domain-specific information is available for both the noun phrase and clause level semantic components through the knowledge base. The domain currently being modelled by SDC is that of computer maintenance reports. Currently the knowledge base is implemented as a semantic net containing a part-whole hierarchy and an **isa** hierarchy of the components and entities in the application domain.

Following the semantic analysis, a discourse component is called which updates the discourse representation to include the information from the current sentence and which runs the focusing algorithm.

2. Uses of Focusing

Focusing is used in four places in PUNDIT -- for definite pronouns, for elided noun phrases, for ***one***-anaphora, and for implicit associates.

As stated above, reference resolution is called by the semantic interpreter when it is in the process of filling a thematic role. Reference resolution proposes a referent for the constituent associated with that role. For example, if the verb is ***replace*** and the semantic interpreter is filling the role of **agent,** reference resolution would be called for the surface syntactic subject. After a proposed referent is chosen for the subject, any specific selectional restrictions on the agent of ***replace*** (such as the constraint that the agent has to be a human being) are checked. If the proposed referent fails selection, backtracking into reference resolution occurs and another referent is selected. Cooperation between reference resolution and the semantic interpreter is discussed in detail in [Palmer1986]. The semantic interpreter itself is discussed in [Palmer1985].

2.1. Pronouns and Elided Noun Phrases

Pronoun resolution is done by instantiating the referent of the pronoun to the first member of the **FocusList** unless the instantiation would violate syntactic constraints on coreferentiality.[3] (As noted above, if the proposed referent fails selection,

[3] At the moment, the syntactic constraints on coreferentiality used by the system are very simple. If the direct object is reflexive it must be instantiated to the same referent as the subject. Otherwise it must be a different referent. Obviously as the system is extended to cover sentences with more complex structures, a more sophisticated treatment of syntactic constraints on

backtracking occurs, and another referent is chosen.)

The reference resolution situation in the maintenance texts however, is complicated by the fact that there are very few overt pronouns. Rather, in contexts where a noun phrase would be expected, there is often elision, or a zero-np as in ***Won't power up*** and ***Has not failed since Hill's arrival.*** Zeroes are handled exactly as if they were pronouns. The hypothesis that elided noun phrases can be treated in the same way as pronouns is consistent with previous claims in [Gundel1980] and [Kameyama1985] that in languages such as Russian and Japanese, which regularly allow zero-np's, the zero corresponds to the focus. If these claims are correct, it is not surprising that in a sublanguage like that found in the maintenance texts, which also allows zero-np's, the zero should correspond to the focus.

Another kind of pronoun (or zero) also occurs in the maintenance texts, which is not associated with the local focus, but is concerned with global aspects of the text. For example, the field engineer is a default agent in the maintenance domain, as in ***Thinks problem is in head select area.*** This is handled by defining ***default elided referents*** for the domain. The referent is instantiated to one of these if no suitable candidate can be found in the **FocusList**.

2.2. Implicit Associates

Focusing is also used in the processing of certain full noun phrases, both definite and indefinite, which involve ***implicit associates.*** The term implicit associates refers to the relationship between ***a disk drive*** and ***the motor*** in examples like ***The field engineer installed a disk drive. The motor failed.*** It is natural for a human reader to infer that the motor is part of the disk drive. In order to capture this intuition, it is necessary for the system to relate the motor to the disk drive of which it is part. Relationships of this kind have been extensively discussed in the literature on definite reference. For example, implicit associates correspond to ***inferrable*** entities described by [Prince1981], the ***associated use definites*** of [Hawkins1978], and the ***associated*** type of implicit backwards specification discussed by [Sidner1979]. Sidner suggests that implicit associates should be found among the entities in focus. Thus, when the system encounters a definite noun phrase mentioned for the first time, it sequentially examines each member of the **FocusList** to determine if it is a possible associate of the current noun phrase. The specific association relationships (such as part-whole, object-property, and so on) are defined in the knowledge base.

This mechanism is also used in the processing of certain indefinite noun phrases. In every domain, it is claimed, there are certain types of entities which can be classified as ***dependent.*** By this is meant an entity which is not typically mentioned on its own, but which is referred to in connection with another entity, on which it is dependent. In the maintenance domain, for example, parts such as keyboards, motors, and printed circuit boards are dependent, since when they are mentioned, they are normally mentioned as being part of something else, such as a console, disk drive, or

coindexing using some of the insights of [Reinhart1976], and [Chomsky1981] will be required.

printer.[4] In an example like *The system is down. The field engineer replaced a bad printed circuit board,* it seems clear that a relationship between the printed circuit board and the system should be represented. Upon encountering a reference to a dependent entity like the printed circuit board, the system looks through the **FocusList** to determine if any previously mentioned entities can be associated with a printed circuit board, and if so, the relationship is made explicit. If no associate has been mentioned, the entity will be associated with a default defined in the knowledge base. For example, in the maintenance domain, parts are defined as dependent entities, and in the absence of an explicitly mentioned associate, they are represented as associated with the system.

2.3. One-Anaphora

PUNDIT extends focusing to the analysis of *one*-anaphora following [Dahl1984], which claims that focus is central to the interpretation of *one*-anaphora. Specifically, the referent of a *one*-anaphoric noun phrase (e.g., *the blue one, some large ones*) is claimed to be a member or members of a set which is the focus of the current clause. For example, in *Installed two disk drives. One failed,* the set of two disk drives is assumed to be the focus of *One failed,* and the disk drive that failed is a member of that set. This analysis can be contrasted with that of [Halliday1976], which treats *one*-anaphora as a surface syntactic phenomenon, completely distinct from reference. It is more consistent with the theoretical discussions of [1976], and [Webber1983]. [5] These analyses advocate a discourse-pragmatic treatment for both *one*-anaphora and definite pronouns. The main computational advantage of treating *one*-anaphora as a discourse problem is that, since definite pronouns are treated this way, little modification is needed to the basic anaphora mechanism to allow it to handle *one*-anaphora. In contrast, an implementation following the account of Halliday and Hasan would be much more complex and specific to *one*-anaphora.

The process of reference resolution for *one*-anaphora occurs in two stages. The first stage is resolution of the anaphor, *one*, and this is the stage that involves focusing. When the system processes the head noun *one*, it instantiates it with the *category* of the first set in the **FocusList** (*disk drive* in this example).[6] In other words, the referent of the noun phrase must be a member of the previously mentioned set of disk drives. The second stage of reference resolution for *one*-anaphora assigns a specific disk drive as the referent of the entire noun phrase, using the same procedures that would be used for a full noun phrase, *a disk drive*.

The extension of the system to *one*-anaphora provides the clearest motivation for the choice of a syntactic focus in PUNDIT. Before I discuss the kinds of examples

---

[4] There are exceptions to this generalization. For example, in a sentence like *field engineer ordered motor*, the motor on order is not part of anything else (yet). In PUNDIT, these cases are assumed to depend on the verb meaning. In this example, the object of *ordered* is categorized as *non-specific*, and reference resolution is not called. See [Palmer1986] for details.

[5] Although not Webber's analysis in [Webber1978], which advocates an approach similar to Halliday and Hasan's.

[6] Currently the only sets in the **FocusList** are those which were explictly mentioned in the text. However, as pointed out by [Dahl1982], and [Webber1983, Dahl1984], other sets besides those explictly mentioned are available for anaphoric reference. These have not yet been added to the system.

which support this approach, I will briefly describe the relevant part of the focusing algorithm based o thematic roles which is proposed by[Sidner1979]. After each sentence, the focusing algorithms order the elements in the sentence in the order in which they are to be considered as potential foci in the next sentence. Sidner's ordering and that of PUNDIT are compared in Figure 1.

The idea that surface syntax is important in focusing comes from a suggestion by [Erteschik-Shir1979], that every sentence has a ***dominant*** syntactic constituent, which provides a default topic for the following utterance[7]. Intuitively, the dominant constituent can be thought of as the one to which the hearer's attention is primarily drawn. Operationally the dominance of a constituent is tested by seeing if a referent with that constituent as the antecedent can be cooperatively referred to with an unstressed pronoun in the following sentence.

The feature of ***one***-anaphora which motivates the syntactic algorithm is that the availability of certain noun phrases as antecedents for ***one***-anaphora is strongly affected by surface word order variations which change syntactic relations, but which do not affect thematic roles. If thematic roles are crucial for focusing, then this pattern would not be observed.

Consider the following examples:

(1) A: I'd like to plug in this lamp, but the bookcases are blocking the electrical outlets.

B: Well, can we move one?

(2) A: I'd like to plug in this lamp, but the electrical outlets are blocked by the bookcases.

Sidner	**PUNDIT**
Theme	**Sentence**
Other thematic roles	**Direct Object**
Agent	**Subject**
Verb Phrase	**Objects of Prepositional Phrases**

Figure 1: Comparison of Potential Focus Ordering in Sidner's System and PUNDIT

[7] As discussed in [Dahl1984] there are problems with Erteschik-Shir's definition of dominance and slightly different definition is proposed. However the details of this reformulation do not concern us here.

B: Well, can we move one?

In (1), most informants report an initial impression that B is talking about moving the electrical outlets. This does not happen for (2). This indicates that the expected focus following (1) A is the outlets, while it is the bookcases in (1) B. However, in each case, the thematic roles are the same, so an algorithm based on thematic roles would predict no difference between (1) and (2).

Similar examples using definite pronouns do not seem to exhibit the same effect. In (3) and (4), *they* seems to be ambiguous, until world knowledge is brought in. Thus, in order to handle definite pronouns alone either algorithm would be adequate.

(3) A: I'd like to plug in this lamp, but bookcases are blocking the electrical outlets.

B: Well, can we move them?

(4) A: I'd like to plug in this lamp, but the electrical outlets are blocked by the bookcases.

B: Well, can we move them?

(5) and (6) illustrate another example with ***one***-anaphora. In (5) but not in (6), the initial interpretation seems to be that a bug has lost its leaves. As in (1) and (2), however, the thematic roles are the same, so a thematic-role-based algorithm would predict no difference between the sentences.

(5) The plants are swarming with the bugs. One's already lost all its leaves.

(6) The bugs are swarming over the plants. One's already lost all its leaves.

In addition to theoretical considerations, there are a number of obvious practical advantages to defining focus on constituents rather than on thematic roles. For example, constituents can often be found more reliably than thematic roles. In addition, thematic roles have to be defined individually for each verb.[8] Since thematic roles for verbs can vary across domains, defining focus on syntax makes it less domain dependent, and hence more portable. While in principle focus based on thematic roles does not have to be domain-dependent, a general algorithm based on thematic roles would have to rely on a a general, domain-neutral specification of all possible thematic roles and their behavior in focusing. Until such a specification exists, a thematic-role based focusing algorithm must be redefined for each new domain as the domain requires the definition of new thematic roles, and because of this, will continue to be less portable than an approach based on syntax.

[8] Of course, some generalizations can be made about how arguments map to thematic roles. For example, the basic definition of the thematic role *theme* is that, for a verb of motion, the theme is the argument that moves. More generally, the theme is the argument that is most affected by the action of the verb, and its typical syntactic manifestation is as a direct object of a transitive verb, or the subject of an intransitive verb. However, even if these generalizations are accurate, they are no more than guidelines for finding the themes of verbs. The verbs still have to be classified individually

3. Implementation

3.1. The FocusList and CurrentContext

The data structures that retain information from sentence to sentence in the PUNDIT system are the **FocusList** and the **CurrentContext**. The **FocusList** is a list of all the discourse entities which are eligible to be considered as foci, listed in the order in which they are to be considered. For example, after a sentence like *The field engineer replaced the disk drive,* the following **FocusList** would be created.

[[event1],[drive1],[engineer1]]

The members of the **FocusList** are unique identifiers that have been assigned to the three discourse entities -- the disk drive, the field engineer, and the event. The **CurrentContext** contains the information that has been conveyed by the discourse so far. After the example above, the **CurrentContext** would contain three types of information:

(1) *Discourse id's*, which represent classifications of entities. For example, **id(field^engineer,[engineer1])** means that **[engineer1]** is a a field engineer.[9]

(2) Facts about part-whole relationships (**hasparts**). In the example in Figure 2, notice that the lack of a representation of time results in both drives being part of the system, which they are, but not at the same time. Work to remedy this problem is in progress.

(3) Representations of the events in the discourse. For example, if the event is that of a disk drive having been replaced, the representation consists of a unique identifier (**[event1]**), the surface verb (**replace(time(_))**), and the decomposition of the verb with its (known) arguments instantiated[10]. The thematic roles involved are **object1,** the replaced disk drive, **object2,** the replacement disk drive, **time** and **instrument** which are uninstantiated, and **agent,** the field engineer. (See[Palmer1986], for details of this representation). Figure 2 illustrates how the **CurrentContext** looks after the discourse-initial sentence, ***The field engineer replaced the disk drive.***

3.2. The Focusing Algorithm

The focusing algorithm used in this system resembles that of [Sidner1979], although it does not use the actor focus and uses surface syntax rather than thematic roles, as discussed above. The focusing algorithm is illustrated in Figure 3. Removing candidates from the **FocusList** when they are no longer eligible to be the referents of pronouns is not currently done in this system. The conditions determining this have not been fully investigated, and since the texts involved are short, few problems are created in practice. This problem will be addressed by future research.

---

[9] **field^engineer** is an example of the representation used in PUNDIT for an idiom.

[10] **_8176** is an uninstantiated variable representing the time of the replacement. It appears in several places, such as **included(object2([drive2]),time(_8176))**, and **missing(object1([drive1]),time(_8176))**.

```
id(field^engineer,[engineer1]),

id(disk^drive,[drive1]),

id(system,[system1]),

id(disk^drive,[drive2]),

id(event,[event1]),


haspart([system1],[drive1]),

haspart([system1],[drive2])]


event([event1],
      replace(time(_8176)),
      [included(object2([drive2]),time(_8176)),
       missing(object1([drive1]),time(_8176)),
       use(instrument(_8405),
          exchange(object1([drive1]),object2([drive2]),time(_8176))),
       cause(agent([engineer1]),
            use(instrument(_8405),
               exchange(object1([drive1]),object2([drive2]),time(_8176))))])
```

Figure 2: CurrentContext after *The field engineer replaced the disk drive.*

(1) **First Sentence of a Discourse:**

Establish expected foci for the next sentence (order FocusList): the order reflects how likely that constituent is to become the focus of the following sentence.

Sentence
Direct Object
Subject
Objects of Prepositional Phrases

(2) **Subsequent Sentences (update FocusList):**

If there is a pronoun in the current sentence, move the focus to the referent of the pronoun. If there is no pronoun, retain the focus from the previous sentence. Order the other elements in the sentence as in (1).

Figure 3: The Focusing Algorithm

4. Summary

Several interesting research issues are raised by this work. For example, what is the source of the focusing algorithm? Is it derivable from theoretical considerations about how language is processed by human beings, or is it simply an empirical observation about conventions used in particular languages to bring discourse entities into prominence? Evidence bearing on this issue would be to what extent the focusing mechanism carries over to other, non-related languages. Kameyama's work on Japanese suggests that there are some similarities across languages. To the extent that such similarities exist, it would suggest that the algorithm is derivable from other theoretical considerations, and is not simply a reflection of linguistic conventions.

This paper has described the reference resolution component of PUNDIT, a large text understanding system in Prolog. A focusing algorithm based on surface syntactic constituents is used in the processing of several different types of reduced reference: definite pronouns, *one*-anaphora, elided noun phrases, and implicit associates. This generality points out the usefulness of treating focusing as a problem in itself rather than simply as a tool for pronoun resolution.

REFERENCES

[1976]
Jorge Hankamer and Ivan Sag, Deep and Surface Anaphora. *Linguistic Inquiry* **7**(3), 1976, pp. 391-428.

[Chomsky1981]
Noam Chomsky, *Lectures on Government and Binding*. Foris Publications, Dordrecht, 1981.

[Dahl1982.]
Deborah A. Dahl, Discourse Structure and one-anaphora in English, presented at the 57th Annual Meeting of the Linguistic Society of America, San Diego, 1982..

[Dahl1984]
Deborah A. Dahl, The Structure and Function of One-Anaphora in English, PhD Thesis; (also published by Indiana University Linguistics Club, 1985), University of Minnesota, 1984.

[Erteschik-Shir1979]
Nomi Erteschik-Shir and Shalom Lappin, Dominance and the Functional Explanation of Island Phenomena. *Theoretical Linguistics*, 1979, pp. 41-86.

[Grosz1983]
Barbara Grosz, Aravind K. Joshi, and Scott Weinstein, Providing a Unified Account of Definite Noun Phrases in Discourse. *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, 1983, pp. 44-50.

[Gruber1976]
Jeffery Gruber, *Lexical Structure in Syntax and Semantics*. North Holland, New York, 1976.

[Gundel1974]
Jeanette K. Gundel, Role of Topic and Comment in Linguistic Theory, Ph.D. thesis, University of Texas at Austin, 1974.

[Gundel1980]
Jeanette K. Gundel, Zero-NP Anaphora in Russian. *Chicago Linguistic Society Parasession on Pronouns and Anaphora*, 1980.

[Halliday1976]
Michael A. K. Halliday and Ruqaiya Hasan, *Cohesion in English*. Longman, London, 1976.

[Hawkins1978]
John A. Hawkins, *Definiteness and Indefiniteness*. Humanities Press, Atlantic Highlands, New Jersey, 1978.

[Hirschman1982]
L. Hirschman and K. Puder, Restriction Grammar in Prolog. In *Proc. of the First International Logic Programming Conference*, M. Van Caneghem (ed.), Association pour la Diffusion et le Developpement de Prolog, Marseilles, 1982, pp. 85-90.

[Hirschman1985]
L. Hirschman and K. Puder, Restriction Grammar: A Prolog Implementation. In *Logic Programming and its Applications*, D.H.D. Warren and M. VanCaneghem (ed.), 1985.

[Hirst1981]
Graeme Hirst, *Anaphora in Natural Language Understanding*. Springer-Verlag, New York, 1981.

[Kameyama1985]
Megumi Kameyama, Zero Anaphora: The Case of Japanese, Ph.D. thesis, Stanford University, 1985.

[Palmer1985]
Martha S. Palmer, Driving Semantics for a Limited Domain, Ph.D. thesis, University of Edinburgh, 1985.

[Palmer1986]
Martha S. Palmer, Deborah A. Dahl, Rebecca J. Schiffman, Lynette Hirschman, Marcia Linebarger, and John Dowding, Recovering Implicit Information, to be presented at the 24th Annual Meeting of the Association for Computational Linguistics, Columbia University, New York, August 1986.

[Prince1981]
Ellen F. Prince, Toward a Taxonomy of Given-New Information. In *Radical Pragmatics*, Peter Cole (ed.), Academic Press, New York, 1981.

[Reinhart1976]
Tanya Reinhart, The Syntactic Domain of Anaphora, Ph.D. thesis, Massachusetts Institute of Technology, 1976.

[Sager1981]
N. Sager, *Natural Language Information Processing: A Computer Grammar of English and Its Applications*. Addison-Wesley, Reading, Mass., 1981.

[Sidner1979]
Candace Lee Sidner, Towards a Computational Theory of Definite Anaphora Comprehension in English Discourse, MIT-AI TR-537, Cambridge, MA, 1979.

[Webber1978]
Bonnie Lynn Webber, *A Formal Approach to Discourse Anaphora*. Garland, New York, 1978.

RECOVERING IMPLICIT INFORMATION

Martha S. Palmer, Deborah A. Dahl, Rebecca J. Schiffman, Lynette Hirschman,
Marcia Linebarger, and John Dowding
Research and Development Division
SDC -- A Burroughs Company
P.O Box 517
Paoli, PA 19301 USA

ABSTRACT

This paper describes the SDC PUNDIT, (Prolog UNDerstands Integrated Text), system for processing natural language messages.[1] PUNDIT, written in Prolog, is a highly modular system consisting of distinct syntactic, semantic and pragmatics components. Each component draws on one or more sets of data, including a lexicon, a broad-coverage grammar of English, semantic verb decompositions, rules mapping between syntactic and semantic constituents, and a domain model.

This paper discusses the communication between the syntactic, semantic and pragmatic modules that is necessary for making implicit linguistic information explicit. The key is letting syntax and semantics recognize missing linguistic entities as implicit entities, so that they can be labelled as such, and reference resolution can be directed to find specific referents for the entities. In this way the task of making implicit linguistic information explicit becomes a subset of the tasks performed by reference resolution. The success of this approach is dependent on marking missing syntactic constituents as elided and missing semantic roles as ESSENTIAL so that reference resolution can know when to look for referents.

To Be Presented at the
24 Annual Meeting of the
Association for Computational
Linguistics
New York City, June, 1986

---

[1] This work is supported in part by DARPA under contract N00014-85-C-0012, administered by the Office of Naval Research. APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED.

1. Introduction

This paper describes the SDC PUNDIT[2] system for processing natural language messages. PUNDIT, written in Prolog, is a highly modular system consisting of distinct syntactic, semantic and pragmatics components. Each component draws on one or more sets of data, including a lexicon, a broad-coverage grammar of English, semantic verb decompositions, rules mapping between syntactic and semantic constituents, and a domain model. PUNDIT has been developed cooperatively with the NYU PROTEUS system (Prototype Text Understanding System), These systems are funded by DARPA as part of the work in natural language understanding for the Strategic Computing Battle Management Program. The PROTEUS/PUNDIT system will map Navy CASREP's (equipment casualty reports) into a database, which is accessed by an expert system to determine overall fleet readiness. PUNDIT has also been applied to the domain of computer maintenance reports, which is discussed here.

The paper focuses on the interaction between the syntactic, semantic and pragmatic modules that is required for the task of making implicit information explicit. We have isolated two types of implicit entities: syntactic entities which are missing syntactic constituents, and semantic entities which are unfilled semantic roles. Some missing entities are optional, and can be ignored. Syntax and semantics have to recognize the OBLIGATORY missing entities and then mark them so that reference resolution knows to find specific referents for those entities, thus making the implicit information explicit. Reference resolution uses two different methods for filling the different types of entities which are also used for general noun phrase reference problems. Implicit syntactic entities, ELIDED CONSTITUENTS, are treated like pronouns, and implicit semantic entities, ESSENTIAL ROLES are treated like definite noun phrases. The pragmatic module as currently implemented consists mainly of a reference resolution component, which is sufficient for the pragmatic issues described in this paper. We are in the process of adding a time module to handle time issues that have arisen during the analysis of the Navy CASREPS.

2. The Syntactic Component

The syntactic component has three parts: the grammar, a parsing mechanism to execute the grammar, and a lexicon. The grammar consists of context-free BNF definitions (currently numbering approximately 80) and associated restrictions (approximately 35). The restrictions enforce context-sensitive well-formedness constraints and, in some cases, apply optimization strategies to prevent unnecessary structure-building. Each of these three parts is described further below.

---

[2] Prolog UNDderstands Integrated Text

2.1. Grammar Coverage

The grammar covers declarative sentences, questions, and sentence fragments. The rules for fragments enable the grammar to parse the "telegraphic" style characteristic of message traffic, such as ***disk drive down,*** and ***has select lock.*** The present grammar parses sentence adjuncts, conjunction, relative clauses, complex complement structures, and a wide variety of nominal structures, including compound nouns, nominalized verbs and embedded clauses.

The syntax produces a detailed surface structure parse of each sentence (where "sentence" is understood to mean the string of words occurring between two periods, whether a full sentence or a fragment). This surface structure is converted into an "intermediate representation" which regularizes the syntactic parse. That is, it eliminates surface structure detail not required for the semantic tasks of enforcing selectional restrictions and developing the final representation of the information content of the sentence. An important part of regularization involves mapping fragment structures onto canonical verb-subject-object patterns, with missing elements flagged. For example, the **tvo** fragment consists of a **tensed verb + object** as in ***Replaced spindle motor.*** Regularization of this fragment, for example, maps the **tvo** syntactic structure into a verb+subject+object structure:

verb(replace),subject(X),object(Y)

As shown here, **verb** becomes instantiated with the surface verb, e.g., ***replace*** while the arguments of the **subject** and **object** terms are variables. The semantic information derived from the noun phrase object ***spindle motor*** becomes associated with Y. The absence of a surface subject constituent results in a lack of semantic information pertaining to X. This lack causes the semantic and pragmatic components to provide a semantic filler for the missing subject using general pragmatic principles and specific domain knowledge.

2.2. Parsing

The grammar uses the Restriction Grammar parsing framework [Hirschman1982, Hirschman1985], which is a logic grammar with facilities for writing and maintaining large grammars. Restriction Grammar is a descendent of Sager's string grammar [Sager1981]. It uses a top-down left-to-right parsing strategy, augmented by dynamic rule pruning for efficient parsing [Dowding1986]. In addition, it uses a meta-grammatical approach to generate definitions for a full range of co-ordinate conjunction structures [Hirschman1986].

2.3. Lexical Processing

The lexicon contains several thousand entries related to the particular subdomain of equipment maintenance. It is a modified version of the LSP lexicon with words classified as to part of speech and subcategorized in limited ways (e.g., verbs are subcategorized for their complement types). It also handles

multi-word idioms, dates, times and part numbers. The lexicon can be expanded by means of an interactive lexical entry program.

The lexical processor reduces morphological variants to a single root form which is stored with each entry. For example, the form ***has*** is transformed to the root form ***have*** in ***Has select lock.*** In addition, this facility is useful in handling abbreviations: the term ***awp*** is regularized to the multi-word expression ***waiting^for^part.*** This expression in turn is regularized to the root form ***wait^for^part*** which takes as a direct object a particular part or part number, as in ***is awp 2155-6147.***

Multi-word expressions, which are typical of jargon in specialized domains, are handled as single lexical items. This includes expressions such as ***disk drive*** or ***select lock***, whose meaning within a particular domain is often not readily computed from its component parts. Handling such frozen expressions as "idioms" reduces parse times and number of ambiguities.

Another feature of the lexical processing is the ease with which special forms (such as part numbers or dates) can be handled. A special "forms grammar", written as a definite clause grammar[Pereira1980] can parse part numbers, as in ***awaiting part 2155-6147***, or complex date and time expressions, as in ***disk drive up at 11/17-1236.*** During parsing, the forms grammar performs a well-formedness check on these expressions and assigns them their appropriate lexical category.

3. Semantics

There are two separate components that perform semantic analysis, NOUN PHRASE SEMANTICS and CLAUSE SEMANTICS. They are each called after parsing the relevant syntactic structure to test semantic well-formedness while producing partial semantic representations. Clause semantics is based on Inference Driven Semantic Analysis [Palmer1985] which decomposes verbs into component meanings and fills their semantic roles with syntactic constituents. A KNOWLEDGE BASE, the formalization of each domain into logical terms, SEMANTIC PREDICATES, is essential for the effective application of Inference Driven Semantic Analysis, and for the final production of a text representation. The result of the semantic analysis is a set of PARTIALLY instantiated semantic predicates which is similar to a frame representation. To produce this representation, the semantic components share access to a knowledge base, the DOMAIN MODEL, that contains generic descriptions of the domain elements corresponding to the lexical entries. The model includes a detailed representation of the types of assemblies that these elements can occur in. The semantic components are designed to work independently of the particular model, and rely on an interface to ensure a well-defined interaction with the domain model. The domain model, noun phrase semantics and clause semantics are all explained in more detail in the following three subsections.

3.1. Domain Model

The domain currently being modelled by SDC is the Maintenance Report domain. The texts being analyzed are actual maintenance reports as they are called into the Burroughs Telephone Tracking System by the field engineers and typed in by the telephone operator. These reports give information about the customer who has the problem, specific symptoms of the problem, any actions take by the field engineer to try and correct the problem, and success or failure of such actions. The goal of the text analysis is to automatically generate a data base of maintenance information that can be used to correlate customers to problems, problem types to machines, and so on.

The first step in building a domain model for maintenance reports is to build a semantic net-like representation of the type of machine involved. The machine in the example text given below is the B4700. The possible parts of a B4700 and the associated properties of these parts can be represented by an **isa** hierarchy and a **haspart** hierarchy. These hierarchies are built using four basic predicates: **system,isa,hasprop, haspart**. For example the system itself is indicated by **system(b4700)**. The **isa** predicate associates TYPES with components, such as **isa(spindle^motor,motor)**. Properties are associated with components using the **hasprop** relationship, are are inherited by anything of the same type. The main components of the system: **cpu**, **power_supply**, **disk**, **printer**, **peripherals**, etc., are indicated by **haspart** relations, such as **haspart(b4700,cpu)**, **haspart(b4700,power_supply)**, **haspart(b4700,disk)**,,etc.. These parts are themselves divided into subparts which are also indicated by **haspart** relations, such as **haspart(power_supply, converter)**.

This method of representation results in a general description of a computer system. Specific machines represent INSTANCES of this general representation. When a particular report is being processed, **id** relations are created by noun phrase semantics to associate the specific computer parts being mentioned with the part descriptions from the general machine representation. So a particular B4700 would be indicated by predicates such as these: **id(b4700,system1)**, **id(cpu,cpu1)**, **id(power_supply,power_supply1)**, etc.

3.2. Noun phrase semantics

Noun phrase semantics is called by the parser during the parse of a sentence, after each noun phrase has been parsed. It relies heavily on the domain model for both determining semantic well-formedness and building partial semantic representations of the noun phrases. For example, in the sentence, *field engineer replaced disk drive at 11/2/0800*, the phrase *disk drive at 11/2/0800* is a syntactically acceptable noun phrase, (as in *participants at the meeting*). However, it is not semantically acceptable in that *at 11/20/800* is intended to designate the time of the replacement, not a

property of the disk drive. Noun phrase semantics will inform the parser that the noun phrase is not semantically acceptable, and the parser can then look for another parse. In order for this capability to be fully utilized, however, an extensive set of domain-specific rules about semantic acceptability is required. At present we have only the minimal set used for the development of the basic mechanism. For example, in the case described here, ***at 11/2/0800*** is excluded as a modifier for ***disk drive*** by a rule that permits only the name of a location as the object of ***at*** in a prepositional phrase modifying a noun phrase.

The second function of noun phrase semantics is to create a semantic representation of the noun phrase, which will later be operated on by reference resolution. For example, the semantics for ***the bad disk drive*** would be represented by the following Prolog clauses.

```
[id(disk^drive,X),
 bad(X),
 def(X),        that is, X was referred to with a full, definite noun phrase,
 full_npe(X)]   rather than a pronoun or indefinite noun phrase.
```

3.3. Clause semantics

In order to produce the correct predicates and the correct instantiations, the verb is first decomposed into a semantic predicate representation appropriate for the domain. The arguments to the predicates constitute the SEMANTIC ROLES of the verb, which are similar to cases. There are domain specific criteria for selecting a range of semantic roles. In this domain the semantic roles include: **agent**,**instrument**,**theme**, **object1**,**object2**, **symptom** and **mod**. Semantic roles can be filled either by a syntactic constituent supplied by a mapping rule or by reference resolution, requiring close cooperation between semantics and reference resolution. Certain semantic roles are categorized as ESSENTIAL, so that pragmatics knows that they need to be filled if there is no syntactic constituent available. The default categorization is NON-ESSENTIAL, which does not require that the role be filled. Other semantic roles are categorized as NON-SPECIFIC or SPECIFIC depending on whether or not the verb requires a specific referent for that semantic role (see Section 4). The example given in Section 5 illustrates the use of both a non-specific semantic role and an essential semantic role. This section explains the decompositions of the verbs relevant to the example, and identifies the important semantic roles.

The decomposition of ***have*** is very domain specific.

```
have(time(Per)) <-
        symptom(object1(O1),symptom(S),time(Per))
```

It indicates that a particular **symptom** is associated with a particular **object**, as in "the disk drive has select lock." The **object1** semantic role

would be filled by the disk drive, the subject of the clause, and the **symptom** semantic role would be filled by *select lock*, the object of the clause. The **time(Per)** is always passed around, and is occasionally filled by a time adjunct, as in *the disk drive had select lock at 0800.*

In addition to the mapping rules that are used to associate syntactic constituents with semantic roles, there are selection restrictions associated with each semantic role. The selection restrictions for *have* test whether or not the filler of the **object1** role is allowed to have the type of symptom that fills the **symptom** role. For example, only disk drives have select locks.

Mapping Rules

The decomposition of *replace* is also a very domain specific decomposition that indicates that an **agent** can use an **instrument** to exchange two **objects**.

```
replace(time(Per)) < -
   cause(agent(A),
      use(instrument(I),
         exchange(object1(O1),object2(O2),time(Per))))
```

The following mapping rule specifies that the **agent** can be indicated by the subject of the clause.

```
agent(A)  < - subject(A)  /  X
```

The mapping rules make use of intuitions about syntactic cues for indicating semantic roles first embodied in the notion of case [Fillmore1968, Palmer1981]. Some of these cues are quite general, while other cues are very verb-specific. The mapping rules can take advantage of generalities like "SUBJECT to AGENT" syntactic cues while still preserving context sensitivities. This is accomplished by making the application of the mapping rules "situation-specific" through the use of PREDICATE ENVIRONMENTS. The previous rule is quite general and can be applied to every **agent** semantic role in this domain. This is indicated by the X on the right hand side of the "/" which refers to the predicate environment of the **agent**, i.e., anything. Other rules, such as "WITH-PP to OBJECT2," are much less general, and can only apply under a set of specific circumstances. The predicate environments for an **object1** and **object2** are specified more explicitly. An **object1** can be the object of the sentence if it is contained in the semantic decomposition of a verb that includes an **agent** and belongs to the *repair* class of verbs. An **object2** can be indicated by a *with* prepositional phrase if it is contained in the semantic decomposition of a replace verb:

```
object1(Part1) < -  obj(Part1)/ cause(agent(A),Repair_event)

object2(Part2) < -
      pp(with,Part2) /
```

cause(agent(A),use(I,exchange(object1(O1),object2(Part2),T)))

Selection Restrictions

The selection restriction on an **agent** is that it must be a field engineer, and an **instrument** must be a tool. The selection restrictions on the two objects are more complicated, since they must be machine parts, have the same type, and yet also be distinct objects. In addition, the first object must already be associated with something else in a **haspart** relationship, in other words it must already be included in an existing assembly. The opposite must be true of the second object: it must not already be included in an assembly, so it must not be associated with anything else in a **haspart** relationship.

There is also a pragmatic restriction associated with both objects that has not been associated with any of the semantic roles mentioned previously. Both **object1** and **object2** are essential semantic roles. Whether or not they are mentioned explicitly in the sentence, they must be filled, preferably by an an entity that has already been mentioned, but if not that, then entities will be created to fill them [Palmer1983]. This is accomplished by making an explicit call to reference resolution to find referents for essential semantic roles, in the same way that reference resolution is called to find the referent of a noun phrase. This is not done for non-essential roles, such as the **agent** and the **instrument** in the same verb decomposition. If they are not mentioned they are simply left unfilled. The **instrument** is rarely mentioned, and the **agent** could easily be left out, as in ***The disk drive was replaced at 0800.***[3] In other domains, the **agent** might be classified as obligatory, and then it wold have to be filled in.

There is another semantic role that has an important pragmatic restriction on it in this example, the **object2** semantic role in ***wait^for^part (awp).***

idiomVerb(wait^for^part,time(Per)) < -
ordered(object1(O1),object2(O2),time(Per))

The semantics of ***wait^for^part*** indicates that a particular type of part has been ordered, and is expected to arrive. But it is not a specific entity that might have already been mentioned. It is a more abstract object, which is indicated by restricting it to being non-specific. This tells reference resolution that although a syntactic constituent, preferably the object, can and should fill this semantic role, and must be of type **machine-part**, that reference resolution should not try to find a specific referent for it (see Section 4).

The last verb representation that is needed for the example is the representation of *be*.

be(time(Per)) < -

---

[3]Note that an elided subject is handled quite differently, as in ***replaced disk drive.*** Then the missing subject is

attribute(theme(T),mod(M),time(Per))

In this domain *be* is used to associate predicate adjectives or nominals with an object, as in ***disk drive is up*** or ***spindle motor is bad.*** The representation merely indicates that a **mod**ifier is associated with an **theme** in an attribute relationship. Noun phrase semantics will eventually produce the same representation for ***the bad spindle motor,*** although it does not yet.

4. Reference Resolution

Reference resolution is the component which keeps track of references to entities in the discourse. It creates labels for entities when they are first directly referred to, or when their existence is implied by the text, and recognizes subsequent references to them. Reference resolution is called from clause semantics when clause semantics is ready to instantiate a semantic role. It is also called from pragmatic restrictions when they specify a referent whose existence is entailed by the meaning of a verb.

The system currently covers many cases of singular and plural noun phrases, pronouns, ***one-*** anaphora, nominalizations, and non-specific noun phrases; reference resolution also handles adjectives, prepositional phrases and possessive pronouns modifying noun phrases. Noun phrases with and without determiners are accepted. Dates, part numbers, and proper names are handled as special cases. Not yet handled are compound nouns, quantified noun phrases, conjoined noun phrases, relative clauses, and possessive nouns.

The general reference resolution mechanism is described in detail in [Dahl1986]. In this paper the focus will be on the interaction between reference resolution and clause semantics. The next two sections will discuss how reference resolution is affected by the different types of semantic roles.

4.1. Obligatory Constituents and Essential Semantic Roles

A slot for a syntactically obligatory constituent such as the subject appears in the intermediate representation whether or not a subject is overtly present in the sentence. It is possible to have such a slot because the absence of a subject is a syntactic fact, and is recognized by the parser. Clause semantics calls reference resolution for such an implicit constituent in the same way that it calls reference resolution for explicit constituents. Reference resolution treats elided noun phrases exactly as it treats pronouns, that is by instantiating them to the first member of a list of potential pronominal referents, the **FocusList.**

---

assumed to fill the **agent** role, and an appropriate referent is found by reference resolution.

The general treatment of pronouns resembles that of[Sidner1979], although there are some important differences, which are discussed in detail in [Dahl1986]. The hypothesis that elided noun phrases can be treated in much the same way as pronouns is consistent with previous claims by [Gundel1980], and [Kameyama1985], that in languages which regularly allow zero-np's, the zero corresponds to the focus. If these claims are correct, it is not surprising that in a sublanguage that allows zero-np's, the zero should also correspond to the focus.

After control returns to clause semantics from reference resolution, semantics checks the selectional restrictions for that referent in that semantic role of that verb. If the selectional restrictions fail, backtracking into reference resolution occurs, and the next candidate on the FocusList is instantiated as the referent. This procedure continues until a referent satisfying the selectional restrictions is found. For example, in ***Disk drive is down. Has select lock***, the system instantiates the disk drive, which at this point is the first member of the **FocusList**, as the **object1** of *have*:

```
[event39]
have(time(time1))
  symptom(object1([drive10]),
        symptom([lock17]),
        time(time1))
```

Essential roles might also not be expressed in the sentence, but their absence cannot be recognized by the parser, since they can be expressed by syntactically optional constituents. For example, in ***the field engineer replaced the motor.***, the new replacement motor is not mentioned, although in this domain it is classified as semantically essential. With verbs like ***replace***, the type of the replacement, ***motor***, in this case, is known because it has to be the same type as the replaced object. Reference resolution for these roles is called by pragmatic rules which apply when there is no overt syntactic constituent to fill a semantic role. Reference resolution treats these referents as if they were full noun phrases without determiners. That is, it searches through the context for a previously mentioned entity of the appropriate type, and if it doesn't find one, it creates a new discourse entity. The motivation for treating these as full noun phrases is simply that there is no reason to expect them to be in focus, as there is for elided noun phrases.

4.2. Noun Phrases in Non-Specific Contexts

Indefinite noun phrases in contexts like ***the field engineer ordered a disk drive*** are generally associated with two readings. In the specific reading the disk drive ordered is a particular disk drive, say, the one sitting on a certain shelf in the warehouse. In the non-specific reading, which is more likely in this

sentence, no particular disk drive is meant; any disk drive of the appropriate type will do. Handling noun phrases in these contexts requires careful integration of the interaction between semantics and reference resolution, because semantics knows about the verbs that create non-specific contexts, and reference resolution knows what to do with noun phrases in these contexts. For these verbs a constraint is associated with the semantics rule for the semantic role **object2** which states that the filler for the **object2** must be non-specific.[4] This constraint is passed to reference resolution, which represents a non-specific noun phrase as having a variable in the place of the pointer, for example, **id(motor,X)**.

Non-specific semantic roles can be illustrated using the **object2** semantic role in *wait^for^part (awp)*. The part that is being *awaited* is non-specific, i.e., can be any part of the appropriate type. This tells reference resolution not to find a specific referent, so the referent argument of the **id** relationship is left as an uninstantiated variable. The analysis of *fe is awp spindle motor* would fill the **object1** semantic role with **fe1** from **id(fe,fe1)**, and the **object2** semantic role with X from **id(spindle^motor,X),** as in **ordered(object1(fe1),object2(X)).** If the spindle motor is referred to later on in a relationship where it must become specific, then reference resolution can instantiate the variable with an appropriate referent such as **spindle^motor3** (See Section 5.6).

5. Sample Text: A sentence-by-sentence analysis

The sample text given below is a slightly emended version of a maintenance report. The parenthetical phrases have been inserted. The following summary of an interactive session with PUNDIT illustrates the mechanisms by which the syntactic, semantic and pragmatic components interact to produce a representation of the text.

1. disk drive (was) down (at) 11/16-2305.
2. (has) select lock.
3. spindle motor is bad.
4. (is) awp spindle motor.
5. (disk drive was) up (at) 11/17-1236.
6. replaced spindle motor.

5.1. Sentence 1: Disk drive was down at 11/16-2305.

As explained in Section 3.2 above, the noun phrase *disk drive* leads to the creation of an **id** *of the form:* **id(disk^drive,[drive1])** Because dates and names generally refer to unique entities rather than to exemplars of a general type, their **ids** do not contain a type argument: **date([11/16-**

[4] The specific reading is not available at present, since it is considered to be unlikely to occur in this domain.

1100]),name([paoli]).

The interpretation of the first sentence of the report depends on the semantic rules for the predicate *be*. The rules for this predicate specify three semantic roles, an **theme** to whom or which is attributed a **modifier**, and the **time**. After a mapping rule in the semantic component of the system instantiates the **theme** semantic role with the sentence subject, ***disk drive,*** the reference resolution component attempts to identify this referent. Because ***disk drive*** is in the first sentence of the discourse, no prior references to this entity can be found. Further, this entity is not presupposed by any prior linguistic expressions. However, in the maintenance domain, when a disk drive is referred to it can be assumed to be part of a B3700 computer system. As the system tries to resolve the reference of the noun phrase ***disk drive*** by looking for previously mentioned disk drives, it finds that the mention of a disk drive presupposes the existence of a system. Since no system has been referred to, a pointer to a system is created at the same time that a pointer to the disk drive is created.

Both entities are now available for future reference. In like fashion, the propositional content of a complete sentence is also made available for future reference. The entities corresponding to propositions are given event labels; thus **event1** is the pointer to the first proposition. The newly created disk drive, system and event entities now appear in the discourse information in the form of a list along with the date.

id(event,[event1])
id(disk^drive,[drive1])
date([11/16-2305])
id(system,[system1])

Note however, that only those entities which have been explicitly mentioned appear in the **FocusList**:

FocusList: [[event1],[drive1],[11/16-2305]]

The propositional entity appears at the head of the focus list followed by the entities mentioned in full noun phrases.[5]

In addition to the representation of the new event, the pragmatic information about the developing discourse now includes information about part-whole relationships, namely that **drive1** is a part which is contained in **system1**.

Part-Whole Relationships:
haspart([system1],[drive1])

The complete representation of **event1**, appearing in the event list in the form shown below, indicates that at the time given in the prepositional phrase ***at 11/16-2305*** there is a state of affairs denoted as **event1** in which a particular

[5] The order in which full noun phrase mentions are added to the **FocusList** depends on their syntactic function and linear order. For full noun phrases, direct object mentions precede subject mentions followed by all other mentions given in the order in which they occur in the sentence. See [Dahl1986], for details.

disk drive, i.e., **drive1**, can be described as *down*.

```
[event1]
   be(time([11/16-2305]))
   attribute(theme([drive1]),
   mod(down),time([11/16-2305]))
```

5.2. Sentence 2: Has select lock.

The second sentence of the input text is a sentence fragment and is recognized as such by the parser. Currently, the only type of fragment which can be parsed can have a missing subject but must have a complete verb phrase. Before semantic analysis, the output of the parse contains, among other things, the following constituent list: **[subj([X]),obj([Y])]**. That is, the syntactic component represents the arguments of the verb as variables. The fact that there was no overt subject can be recognized by the absence of semantic information associated with **X**, as discussed in Section 3.2. The semantics for the maintenance domain sublanguage specifies that the thematic role instantiated by the direct object of the verb *to have* must be a symptom of the entity referred to by the subject. Reference resolution treats an empty subject much like a pronominal reference, that is, it proposes the first element in the **FocusList** as a possible referent. The first proposed referent, **event1** is rejected by the semantic selectional constraints associated with the verb *have*, which, for this domain, require the role mapped onto the subject to be classified as a machine part and the role mapped onto the direct object to be classified as a symptom. Since the next item in the **FocusList**, **drive1**, is a machine part, it passes the selectional constraint and becomes matched with the empty subject of *has select lock*. Since no select lock has been mentioned previously, the system creates one. For the sentence as a whole then, two entities are newly created: the select lock (**[lock1]**) and the new propositional event (**[event2]**): **id(event,[event2]), id(select^lock,[lock1])**. The following representation is added to the event list, and the **FocusList** and **Ids** are updated appropriately.[6]

```
[event2]
have(time(time1))
symptom(object1([drive1]),
symptom([lock1]),time(time1))
```

5.3. Sentence 3: Motor is bad.

In the third sentence of the sample text, a new entity is mentioned, *motor*. Like *disk drive* from sentence 1, *motor* is a dependent entity. However, the entity it presupposes is not a computer system, but rather, a disk drive. The

---

[6] This version only deals with explicit mentions of time, so for this sentence the time argument is filled in with a gensym that stands for an unknown time period. The current version of PUNDIT uses verb tense and verb semantics

newly mentioned motor becomes associated with the previously mentioned disk drive.

After processing this sentence, the new entity **motor3** is added to the **FocusList** along with the new proposition **event3**. Now the discourse information about part-whole relationships contains information about both dependent entities, namely that **motor1** is a part of **drive1** and that **drive1** is a part of **system1**.

```
haspart([drive1],[motor1])
haspart([system1],[drive1])
```

5.4. Sentence 4: is awp spindle motor.

Awp is an abbreviation for an idiom specific to this domain, ***awaiting part.*** It has two semantic roles, one of which maps to the sentence subject. The second maps to the direct object, which in this case is the non-specific spindle motor as explained in Section 4.2. The selectional restriction that the first semantic role of ***awp*** be an engineer causes the reference resolution component to create a new engineer entity because no engineer has been mentioned previously. After processing this sentence, the list of available entities has been incremented by three:

```
id(event,[event4])
id(part,[_2317])
id(field^engineer,[engineer1])
```

The new event is represented as follows:

```
[event4]
idiomVerb(wait^for^part,time(time2))
wait(object1([engineer1]),
    object2([_2317]),time(time2))
```

5.5. Sentence 5: disk drive was up at 11/17-0800 In the emended version of sentence 5 the ***disk drive*** is presumed to be the same drive referred to previously, that is, **drive1**. The semantic analysis of sentence 5 is very similar to that of sentence 1. As shown in the following event representation, the predicate expressed by the modifier ***up*** is attributed to the theme **drive1** at the specified time.

```
[event5]
be(time([11/17-1236]))
attribute(theme([drive1]),
mod(up),time([11/17-1236]))
```

---

to derive implicit time arguments.

5.6. Sentence 6: Replaced motor.

The sixth sentence is another fragment consisting of a verb phrase with no subject. As before, reference resolution tries to find a referent in the current **FocusList** which is a semantically acceptable subject given the thematic structure of the verb and the domain-specific selectional restrictions associated with them. The thematic structure of the verb *replace* includes an **agent** role to be mapped onto the sentence subject. The only **agent** in the maintenance domain is a field engineer. Reference resolution finds the previously mentioned engineer created for *awp spindle motor,* **[engineer1].** It does not find an **instrument,** and since this is not an essential role, this is not a problem. It simply fills it in with another gensym that stands for an unknown filler, **unknown1.**

When looking for the referent of a spindle motor to fill the **object1** role, it first finds the non-specific spindle motor also mentioned in the *awp spindle motor* sentence, and a specific referent is found for it. However, this fails the selection restrictions, since although it is a machine part, it is not already associated with an assembly, so backtracking occurs and the referent instantiation is undone. The next spindle motor on the **FocusList** is the one from *spindle motor is bad,* ([**motor1**]). This does pass the selection restrictions since it participates in a **haspart** relationship.

The last semantic role to be filled is the **object2** role. Now there is a restriction saying this role must be filled by a machine part of the same type as **object1,** which is not already included in an assembly, viz., the non-specific spindle motor. Reference resolution finds a new referent for it, which automatically instantiates the variable in the **id** term as well. The representation can be decomposed further into the two semantic predicates **missing** and **included**, which indicate the current status of the parts with respect to any existing assemblies. The **haspart** relationships are updated, with the old **haspart** relationship for [**motor1**] being removed, and a new **haspart** relationship for [**motor3**] being added. The final representation of the text will be passed through a filter so that it can be suitably modified for inclusion in a database.

```
[event6]
 replace(time(time3))
 cause(agent([engineer1]),
    use(instrument(unknown1),
        exchange(object1([motor1]),
                 object2([motor2]),
                 time(time3))))
 included(object2([motor2]),time(time3))
 missing(object1([motor1]),time(time3))
```

Part-Whole Relationships:

```
haspart([drive1],[motor3])
haspart([system1],[drive1])
```

6. Conclusion

This paper has discussed the communication between syntactic, semantic and pragmatic modules that is necessary for making implicit linguistic information explicit. The key is letting syntax and semantics recognize missing linguistic entities as implicit entities, so that they can be marked as such, and reference resolution can be directed to find specific referents for the entities. Implicit entities may be either empty syntactic constituents in sentence fragments or unfilled semantic roles associated with domain-specific verb decompositions. In this way the task of making implicit information explicit becomes a subset of the tasks performed by reference resolution. The success of this approach is dependent on the use of syntactic and semantic categorizations such as ELLIDED and ESSENTIAL which are meaningful to reference resolution, and which can guide reference resolution's decision making process.

ACKNOWLEDGEMENTS

We would like to thank Bonnie Webber for her very helpful suggestions on exemplifying semantics/pragmatics cooperation.

REFERENCES

[Dahl1986]
Deborah A. Dahl, Focusing and Reference Resolution in PUNDIT, submitted for publication, 1986.

[Dowding1986]
John Dowding and Lynette Hirschman, Dynamic Translation for Rule Pruning in Restriction Grammar, submitted to AAAI-86, Philadelphia, 1986.

[Fillmore1968]
C. J. Fillmore, The Case for Case. In ***Universals in Linguistic Theory***, E. Bach and R. T. Harms (ed.), Holt, Rinehart, and Winston, New York, 1968.

[Gundel1980]
Jeanette K. Gundel, Zero-NP Anaphora in Russian. ***Chicago Linguistic Society Parasession on Pronouns and Anaphora***, 1980.

[Hirschman1982]
L. Hirschman and K. Puder, Restriction Grammar in Prolog. In ***Proc. of the First International Logic Programming Conference***, M. Van Caneghem (ed.), Association pour la Diffusion et le Developpement de Prolog, Marseilles, 1982. pp. 85-90.

[Hirschman1985]
L. Hirschman and K. Puder, Restriction Grammar: A Prolog Implementation. In ***Logic Programming and its Applications***, D.H.D. Warren and M. VanCaneghem (ed.), 1985.

[Hirschman1986]
L. Hirschman, Conjunction in Meta-Restriction Grammar. ***J. of Logic Programming***, 1986.

[Kameyama1985]
Megumi Kameyama, Zero Anaphora: The Case of Japanese, Ph.D. thesis, Stanford University, 1985.

[Palmer1983]
M. Palmer, Inference Driven Semantic Analysis. In ***Proceedings of the National Conference on Artificial Intelligence (AAAI-83)***, Washington, D.C., 1983.

[Palmer1981]
Martha S. Palmer, A Case for Rule Driven Semantic Processing. ***Proc. of the 19th ACL Conference***, June, 1981.

[Palmer1985]
Martha S. Palmer, Driving Semantics for a Limited Domain, Ph.D. thesis, University of Edinburgh, 1985.

[Pereira1980]
F. C. N. Pereira and D. H. D. Warren. Definite Clause Grammars for Language Analysis -- A Survey of the Formalism and a Comparison with Augmented Transition Networks. ***Artificial Intelligence* 13**, 1980, pp. 231-278.

[Sager1981]
N. Sager, ***Natural Language Information Processing: A Computer Grammar of English and Its Applications.*** Addison-Wesley, Reading, Mass., 1981.

[Sidner1979]
Candace Lee Sidner, Towards a Computational Theory of Definite Anaphora Comprehension in English Discourse, MIT-AI TR-537, Cambridge, MA, 1979.

DESIGNING LEXICAL ENTRIES FOR A LIMITED DOMAIN

April, 1986

Rebecca J. Schiffman
SDC--A Burroughs Company
Paoli, PA

1. Introduction

This report outlines procedures for building domain specific lexical entries for the PUNDIT natural language system at SDC. The lexical entries are designed for utilization in inference-driven semantic analysis (Palmer, 1984). The procedures for constructing the lexical entries take advantage of recent works in linguistic semantics (cf. **References Cited**, esp. Dowty, 1979; Foley and Van Valin, 1984; Levin, 1985; Levin and Rappaport, 1985; Rappaport and Levin, 1985; and Talmy, 1978a, 1978b, 1985) without being constrained by any particular linguistic theory. Of particular utility is a section in Foley and Van Valin (1984) entitled "The Semantic Structure of the Clause" in which they draw on the work of Gruber (1965), Jackendoff (1976) and Dowty (1979). Their aim is to provide a set of general tools for the semantic analysis of the verb system of any language. The generality of their approach makes it appropriate not only for different languages but also for domain-specific sub-languages.

This is the first report in a series of two on designing lexical entries. It gives an overview of the general methods for constructing lexical entries regardless of the domain. A subsequent report will focus on specific semantic issues pertaining to the current domain application of PUNDIT. This domain consists of Navy casualty reports (casreps) describing failures in shipboard starting air compressors (sacs).

2. General

The lexical entries consist of predicate logic clauses which represent word meaning and thematic structure in a single decomposition. Currently, two classes of words are given lexical entries: 1) those that serve as predicates (excluding predicate nominals[1]) i.e., verbs, adjectives and prepositions, and 2) deverbal nouns and other nouns which take arguments.[2] Predicating expressions can be classified on the basis of similarities of meaning and thematic structure, and the similarities can then be captured by assigning similar predicate structures to classes of expressions. The predicate structures comprising the lexical entries for the casreps contain three types of abstract elements: basic semantic predicates (primitives), thematic roles, and aspectual operators.

The three elements of a lexical decomposition are all represented as predicate-argument terms embedded in a semantic tree structure, but they have distinct functions. The thematic role predicates, e.g., **agent** and **patient**, are the leaves of the semantic tree whose arguments are constituents of surface structure (e.g., subject, direct object). Thus each role type has an associated set of possible mappings to surface structure (e.g., an **agent** can be realized as a subject or as the object of a *by* phrase). Thematic roles are in turn the arguments of superordinate semantic predicates, the semantic primitives in terms of which the lexical content of a predicate is represented. The aspectual operators represent the temporal structure of a predicating expression and are necessarily superordinate to one or more semantic primitives.

---

[1]Nominals occur in a variety of predicational uses, e.g., equational sentences (e.g., *Scott is the author of Waverly*) and sentences expressing type relations (e.g., *A persimmon is a (type of) fruit*). One way to represent such sentences would be to fill in a variable of a pre-defined predicate provided in the knowledge domain: e.g., **author(Scott,Waverly)**, and **isa(persimmon,fruit)**.

[2]Chomsky (1970) gives a short list of nouns with various complements, many but not all of which would fall into the category of nouns with thematic structure. Levi (1978) relates the complements of such nouns to 'semantically based case relations' (p. 27). Nouns in the current domain which take arguments include those classified as 'percepts', e.g., *color* as in *color of oil*; also, those classified as 'scalars', e.g., *pressure* as in *lube oil pressure of 65 psi*.

Decomposition Structure of BREAK:

a) Semantic roles appear in italics
b) Semantic predicates are capitalized
c) Aspectual operator appears in boldface

CAUSE(*agent(_)*,**become**(BROKEN(*patient(_)*)))

While lexical entries are necessarily domain specific, there are general principles which can guide the determination of all three components.

Lexical content, thematic structure and inherent aspect can be distinguished conceptually, but have complex (lattice-like) interdependencies. Regardless of which type of semantic component motivates the preliminary classification of expressions in a domain, the sub-classes will cut across categories. For example, agents are associated with two distinct aspectual classes, activity and event predications. Thus, arriving at a semantic classification of a set of predicating expressions is a cyclic rather than linear task.

3. Basic Semantic Predicates

Given an existing knowledge base, the domain specific semantic primitives could be selected to accord with relations specified in the knowledge base. In the absence of an a priori set of semantic relations, semantic classes can be chosen by grouping predicating expressions on the basis of general meaning classes, e.g., verbs indicating change of location (*move*), manner of motion (*slide*), change of physical state (*melt*), cognition (*suspect*), and so on. The actual decompositions within a class of expressions would depend on how accurately the meaning of the expressions must be represented. Thus selecting the semantic primitives for a domain depends largely on the application.

4. Aspect

Talmy provides a concise definition of aspect as 'the pattern of distribution of action through time' and observes that a particular aspectual content is generally part of the inherent meaning of a verb, though this inherent meaning can be modified by grammatical elements with aspectual meaning. Representing aspect in lexical entries makes it possible to appropriately interpret tense, grammatical aspect (i.e., progressive) and temporal phrases. The number of aspectual distinctions proposed in analyses of lexical aspect varies, depending on the language being investigated and the predilections of the investigator, but the minimal set consists of the distinction between stative and non-stative predications, and for the latter, between activities and events (change-of-state or change-of-location predications). Stative predications denote states of affairs which persist throughout some period of time during which there is no change or activity, i.e., the truth of the predication can be determined by sampling the state of affairs at a single point in time. Activity predications also denote states of affairs which persist for some period of time but differ from statives in that some activity or process is ensuing such that there is change from moment to moment. Event predications denote a transition to a new state of affairs, e.g., into a new physical state (*The ice melted*) or to a new location (*The ship arrived in port*).

4.1. Diagnostics and defining criteria

A variety of semantic criteria and sentence frames have been proposed to distinguish between aspectual classes (cf. Dowty, 1979). Since only three aspectual classes are implemented in PUNDIT, identifying two of them--statives and events--is sufficient. Activity predications are then predications which are neither states nor events.

Statives

a) cannot be referenced with *do it* (not applicable with passive voice)

Event: *The oil sometimes ignites;*
it does it when the oil pressure is too high.

State: * *The oil is sometimes dark in color;*
it does it when the oil pressure is too high.

b) cannot occur in pseudo-clefts: *what X did was Y*

Event: *What the oil did was ignite.*
State: * *What the oil did was be dark.*

c) nominalization of whole VP cannot be subject of *occur, take place*

Event: *The oil's igniting occurs too frequently.*
State: * *The oil's being dark takes place twice a day.*

Events

a) the past participle of change of state (event) predicates can be used adjectivally; e.g., the surface sentence "NP is V-ed"is more likely to be interpreted as a stative predication than as an event expressed in the passive voice

NP is [activity verb]-ed tends to be interpreted as a recurrent event: *The engine is* [usually] *operated*

NP is [event verb]-ed tends to be interpreted as a current state: *The engine is* [now] *corroded*

b) a sentence in the past tense entails that the patient or theme is in a new state or new location

New location: *The ship arrived in port at 1300 hours.*
Entails: **The ship is in port as of 1300 hours**

c) past progressive predication does not entail the simple past

Activity predication:
The engineer was operating the machinery.
Entails:
The machinery operated.

Event predication:
The crew was installing a new engine
Does not entail:
The crew installed a new engine.

4.2. Representation

Following Dowty (1979) and Foley and Van Valin (1984), the aspectual meaning of predicating expressions is represented in part in their decompositional structure. Event decompositions contain a become predicate. The resulting state or location of an event verb is embedded directly beneath the become predicate, e.g., *fail* is represented as become(failed(_)). Currently,

distinguishing states from activities is not done via an aspectual operator. In the current domain, stative predications (excluding those treated as "transparent" predicates, e.g., cognition verbs) are those whose main verb is *be* or *have* (e.g., *be inoperative*; *have wear*). All other non-event verbs are activities. For domains with a more heterogeneous class of stative predications, an aspectual operator (e.g., Dowty's **do**) could be added to activity decompositions to distinguish them from statives in future implementations.

More fine grained treatments of lexical aspect distinguish between types of activities and types of events. For example, Talmy (1985) classifies activities into full-cycle (*strike*), multiplex (*breathe*), and steady-state (*sleep*). His distinction between full-cycle and steady-state corresponds roughly to the more familiar terminological distinction between punctual and non-punctual verbs. A full-cycle predication can be transformed into a multiplex when a duration is associated with the activity. The duration adverbial forces an interpretation of repeated instances throughout the duration (e.g., *someone struck the gong*=one **strike-gong** event; versus *someone struck the gong for three hours*=repeated **strike-gong** events). Because such distinctions can affect the interpretation of adverbial expressions, future domain applications might benefit from a fine-grained typology of activities. In the current application, activities are not subcategorized.

Causation is generally treated in discussions of aspect because causal predications are necessarily temporally complex: an activity of one participant causes a resulting state or activity in another participant. In other words, the logical structure of a causative verb can be represented as **cause(predicate1(agent(_)),predicate2(*role*(_))**. **Predicate1** generally, if not always, falls into the aspectual class of activities, whereas **predicate2** may be either an activity or a simple event. The crucial component of the first term in a **cause** predicate is the **agent** semantic role. For notational simplicity, **agent(_)** can be substituted for **predicate1(agent(_))** without obscuring the distinction between the two aspectually distinct types of causatives. The general decompositional structure for causatives resulting in an activity is thus: **cause(agent(_),Pred(actor(_)))** (e.g., *someone operated the sac* <- **cause(agent(_),operate(actor(sac)))**). Causatives resulting in a new state or location are represented as: **cause(agent(_),become(Pred(patient(_))))** or **cause(agent(_),become(Pred(theme(_),location(_))))** (e.g., *the drive shaft sheared the driven gear* <- **cause(actor(drive shaft),become(sheared(patient(driven gear))))** where **become** is embedded in the decomposition). Aspectual operators also have relevance to thematic structure as will be shown in the following section.

5. Thematic structure

There is no a priori set of thematic roles with fixed criteria for assigning the arguments of a predication to one or another role type. However, there are gross regularities in the lexicon pertaining to 1) the number of arguments a verb takes in various uses (e.g., transitive/intransitive uses of the same morphological form), 2) the syntactic relations between the verb and its arguments, 3) and the interpretation of how an argument participates in the state, activity or event expressed in the predication. All three factors contribute to the analysis of thematic structure. The following discussion outlines a procedure for assigning thematic structure.

The distinction between stative and event predications and the discussion of causation provide a starting point for determining thematic structure in the following ways. First, all event predications, by definition, contain stative predications within them, i.e., all event predications are either of the form become(stative), if intransitive (e.g., *the sac failed*), or **cause(X,become(stative))** if transitive (e.g., *the operator disengaged the sac*). The aspectual operator become doesn't change the thematic structure of a predicate. In contrast, the **cause** predicate is both an indication of causative meaning and of the presence of an agent thematic role. There is thus a regular relationship between the thematic structure and valency of a stative predication (**NP1 be X**), a simple-event whose result is the stative predication (**NP1 become X**), and the related causative-event (**NP2 cause NP become X**). For any stative, there may or may not be a corresponding intransitive predication: *the cup is broken/the cup broke* versus *the*

*drive shaft is lubricated/ *the drive shaft lubricated.* Further, the event and stative predications may or may not make use of morphologically related forms. A first pass at determining the set of thematic roles associated with the predications used in a particular domain can be accomplished by examining triplets of stative/simple-event/causative-event predicates on the one hand, and pairs of simple activity/causative-activity predicates on the other.

5.1. Predications with Patient/Theme Arguments

A large number of event predications fall into one of two classes: state-change or location-change. The argument said to undergo a change of state is conventionally a **patient** while one said to undergo a change of location is conventionally a **theme**. The state-change state predicates typically have only the **patient** role while location-change predicates typically involve at least one location role (e.g., **source** and/or **goal**). Further, both **patients** and **themes** tend to be subjects of simple event predication and direct objects of causative events. Corresponding to these two types of event predications are two types of stative predications specifying the current state or current location of an entity. The two types of stative predications, which tend to be of the form *NP is Adj* or *NP is locative-PP*, have the same semantic roles as their corresponding event predications. The following chart schematically represents the three aspectual types--stative, simple-event and causative-event--of the two semantic classes--location and physical state:

Stative predication:
  Physical state. "the shaft is dry"
    <- dry(patient(shaft))
  Location: "metal particles are in the oil"
    <- in(theme(particles),location(oil))
Simple event:
  Physical state: "the pump seized"
    <- become(seized(patient(pump)))
  Location: "the ship arrived at the port"
    <- become(at(theme(ship),location(port)))
Causative event:
  Physical state: "the operator disengaged the sac"
    <- cause(agent(operator), become(disengaged(patient(sac))))
  Location: "the operator disconnected the shaft from the hub"
    <- cause(agent(operator),
        become(disconnected(theme(shaft),location(hub))))

Fig 1. Six abstract semantic types

Other roles in addition to **agent, patient, theme** and **location** are sometimes associated with stative and event predications. For example, a causative event verb may have an **instrument** role, depending in part on whether an inanimate entity can be the subject of the causative transitive, as in *the hammer broke the cup.* As mentioned above, change of location verbs may have **source** or **goal** roles. Whether to incorporate an **instrument** role, or to substitute **source** or **goal** for **location**, depends in part on what arguments can appear in surface structure and on the set of semantic primitives appropriate for the domain. For example, the **location** argument of *disconnect* is more precisely a **source** as evidenced by the possibility of a *from* prepositional phrase alongside the impossibility of a *to* phrase:

*the operator disconnected the shaft from/ *to the hub*

Other change of location verbs may take both **goal** and **source**, or only **goal**:

the ship went from the harbor to the open sea
*the operator attached the shaft to/ *from the hub*

Both **sources** and **goals** are types of **locations**. Their contribution to lexical meaning can be captured by the choice of thematic roles or by the choice of semantic primitives. Thus the **location** argument of *disconnect* could be represented as a **source**: **disconnect(theme,source)**. Alternatively, the meaning captured by the **source** role, viz. that the **theme** is no longer at some source location, could be represented by embedding a **location** role in the negation of an *at* predicate:

disconnect <- become(not(at(theme(_),location(_)))).

Similarly, the logical structure of *the ship went from the harbor to the open sea* could be represented in a relatively flat, or inferentially shallow structure, as in:

move(theme(ship),source(harbor),goal(sea)).

Alternatively, the lexical decomposition process could be carried a step further to incorporate the logical inferences represented below (cf. Foley and Van Valin, pp. 51ff):

at(theme(ship),location(sea)),
not(at(theme(ship),location(harbor))).

This is a very simple illustration of how the set of thematic roles for a domain interacts with the set of primitive semantic predicates, which in turn depends on the desired output structures. The choice between implementing only a **location** role for a domain, or all three **location, source** and **goal** roles, also affects the set of surface structure mappings for locative arguments.

5.2. Actor predications

An activity predication minimally requires an argument which is the entity performing an act or engaged in some process, here called the **actor**. Thus **actors** are generally animate entities, or inanimate entities which have a source of energy or motive force. Examples of activity predications taking only an **actor** argument are:

the woman sneezed <- **sneeze(actor(woman))**
the wind blew <- **blow(actor(wind))**
the wheel turned <- **turn(actor(wheel))**

Some activity predications of this form also have transitive/causative uses and in effect have two **actor** roles, a causing **actor** and an experiencing **actor**. The former is designated an **agent**, as in:

someone turned the wheel <- **cause(agent(someone),turn(actor(wheel)))**.

The verb *turn* illustrates a relationship between a univalent activity predicate and its corresponding bivalent causative. Not all bivalent activity predicates are causatives in this sense. There are some transitive activity verbs whose direct object argument is not an **actor**, but rather, a passive participant, e.g., a **theme** as in:

someone kicked the wall <- **kick(actor(someone),theme(wall))**.

In sum, most activity predicates can be classified as one of the three following types:

Activity predication:
  Univalent: **Pred(actor(_))**
  Bivalent causative: **Cause(agent(_),Pred(actor(_)))**
  Bivalent non-causative: **Pred(actor(_),theme(_))**
    or: **Pred(actor(_),location(_))**.

Fig. 2. Four abstract semantic types

6. Summary of simple predicate types

The following chart, which amalgamates Figs. 1 and 2 above, schematizes classes of predicates by valency, general thematic type and aspectual class.

Stative predication:
  state Pred(patient(_))
  location Pred(theme(_),location(_))
Simple-event predication:
  change of state become(Pred(patient(_)))
  change of location become(Pred(theme(_),location(_)))
Causative-event predication:
  Physical state: cause(agent(_),become(Pred(patient(_))))
  Location: cause(agent(become(Pred(theme(_),location(_)))))
Activity predication:
  univalent Pred(actor(_))
  bivalent,
    non-causative Pred(actor(_),theme(_))
      or Pred(actor(_),location(_))
    causative cause(agent(_),Pred(actor(_)))

Fig. 3. Ten abstract semantic types

Patients and **themes** are both associated with stative and simple event predications: **patients** are associated with predicates characterizing the physical state of some entity (or state-change) while **Themes**, together with **locations**, are associated with predicates describing the location of some entity (or location-change). **Patients** and **themes** are also alike in having similar surface

structure realizations; both are subjects of stative or intransitive predications or direct objects of transitive-causatives. The presence of a **become** operator in a decomposition changes the aspect of a predicate from stative to simple-event without changing the valency. **Actors** are associated with activity predicates, which may be inherently intransitive or transitive. For transitive activity predications, the second argument is likely to be a location or a theme. The **agent** role invariably indicates a causative predication, of which there are two aspectual types: causative-events and causative-activities. In a causative-event, the agent causes some entity to enter a new state or location; in a causative-activity, the agent causes some entity to engage in a new activity. Often a causative predication and the corresponding simple-event or activity are expressed by the same morphological form (cf. *turn*).

As shown above, the thematic roles built into a decomposition reflect in part the aspectual properties and valency of a surface predicate as well as the distinction between state-change and location-change meaning. It has been briefly observed that in addition, each thematic role has certain prototypical surface realizations. These are reviewed in greater detail in the next section.

7. Mappings from thematic structure to surface structure

The most salient arguments of a predicating expression are those appearing as clausal subjects and direct objects. Predicating expressions can also occur in noun phrases, e.g., adjectives and prepositional phrases. The following chart summarizes the typical surface realizations in both noun phrases and basic clauses of the thematic roles reviewed above, except for **location**. As the earlier discussion of the verb *disconnect* suggests, some change of location verbs are inherently directional (*disconnect from/ *to*, compare *put on* versus *take off*). Others are not, and thus take a wide variety of locative complements (e.g., *move to/from/by; pass in/out/by*). Motion verbs (in English, cf. Talmy) tend to incorporate manner and cause as well as simple motion (e.g., *stand, bounce, hang, twist, pull* and so on). For these and other reasons, the surface realizations of **location** arguments are more idiosyncratic than the other arguments reviewed above. Discussion of **location** arguments will be postponed.

CHART OF THEMATIC ROLE TO SURFACE STRUCTURE MAPPINGS

AGENT IS REALIZED AS:

1) Possessive determiner of gerund/nominalization:
'the engineer's replacement of the sac'
'the engineer's replacing the sac'

2) Subject of finite or non-finite clause:
'the engineer replaced/replacing the sac'

3) PP obj of 'by' in a passive:
'the sac was replaced by the engineer'

PATIENT IS REALIZED AS:

1) Noun modifiying a nominalization:
'sac disengagement'
'impeller blade tip erosion'

2) PP obj of 'of', where head is gerund, nominalization or related noun:
'disengaging of sac'
'disengagement of sac'
'erosion of impellor blade tip'

3) Possessive determiner of gerund/nominalization:
"sac's disengagement"

4) Head of NP where left modifier is adj or pple
requiring patient role:
'broken tooth'
'burnt odor'

5) Subject of copula/passive S:
'gear teeth are broken'
'oil is discolored'

6) Direct object of transitive: 'the operator broke the sac'

7) Subject of intransitive, if it exists: : 'the gear tooth broke'

THEME IS REALIZED AS:

1) PP obj of 'of' for nominalization/gerund:
'disconnection of coupling'
'color of oil'

2) Head of NP whose left modifier is a pred requiring a theme:
'packed drive shaft'
'disconnected shaft'

3) Subject of copula/passive S:
'drive shaft was packed'
'shaft was disconnected'

4) Dobj of causative tr.:
'someone packed the drive shaft'
'someone disconnected the diesel hub'

ACTOR IS REALIZED AS:

1) PP obj of 'of' for gerund or nominalization:
   'sounding of alarm'
   'rotation of drive shaft '

2) Possessive determiner of gerund/nominalization:
   'the alarm's sounding'

3) Noun modifying a nominalization:
   'engine operation'

4) Subject of intransitive:
   'the alarm sounded'
   'the drive shaft rotated'

5) Subject of passive S:
   'drive shaft was rotating'
   'engine was operated'

6) Dobj of causative:
   'someone sounded the alarm'

References Cited

Dowty, David R. 1979. Word Meaning and Montague Grammar: The Semantics of Verbs and Times in Generative Semantics and in Montague's PTQ. Dordrecht: D. Reidel.

Foley, William and Van Valin, R. 1984. Functional Syntax and Universal Grammar. Cambridge: Cambridge University Press.

Levin, Beth, ed. 1985. Lexical semantics in review. Lexicon Project Working Papers no. 1. Center for Cognitive Science. Cambridge, MA: MIT.

Levin, Beth and Malka Rappaport. 1985. The formation of adjectival passives. Lexicon Project Working Papers no. 2. ms.

Rappaport, Malka and Beth Levin. 1985. A case study in lexical analysis: the locative alternation. ms.

Talmy, Leonard. 1985. Lexicalization patterns: semantic structure in lexical forms. In Language Typology and Syntactic Description, vol. 3: Grammatical Categories and the Lexicon, pp. 57-151. Edited by Timothy Shopen. Cambridge: Cambridge University Press.

Talmy, Leonard. 1978a. Figure and ground in complex sentences. In Universals of Human Language, vol. 4, pp. 627-49. Edited by Charles Ferguson and Edith Moravcsik. Stanford: Stanford University Press.

Talmy, Leonard. 1978b. Relations between subordination and coordination. In Universals of Human Language, vol. 4, pp. 489-513. Edited by Charles Ferguson and Edith Moravcsik. Stanford: Stanford University Press.

Grammatical coverage of the CASREPS:
Summary of current status
April, 1986

Marcia Linebarger

1. COVERAGE OF CASREPS

Total of sentences: 154
- Total parsed correctly: 131 (85%)
  - On 1st, 2nd, or 3rd parse: 109
    - On 1st parse: 92
    - On 2nd or 3rd parse: 17
  - On 4th or subsequent parse: 22
- Total not parsed at all, or parsed incorrectly: 23
  - Due to ill-formed input: 9
  - Due to lexical scanner problems: 7
  - Due to inadequacies of grammar coverage: 4
  - Due to xor (correct reading available but not generated): 3

The figures below represent coverage of the same corpus with the lexical scanner difficulties revolved and the ill-formed input (misspellings, mispunctuations, run-on sentences) corrected. Since two of these sentences would need to be re-phrased in order to be corrected, they are simply omitted from the sentence total in the following breakdown:

Total of sentences (less two): 152
- Total parsed correctly: 145 (95%)
  - On 1st, 2nd, or 3rd parse: 120
    - On 1st parse: 101
    - On 2nd or 3rd parse: 19
  - On 4th or subsequent parse: 25
- Total not parsed at all, or parsed incorrectly: 7
  - Due to inadequacies of grammar coverage: 4
  - Due to xor (correct reading available but not generated): 3

2. EXTENSIONS TO GRAMMAR

The extensions to the grammar required to parse this corpus include the addition of rules for fragments, objects, sentence adjuncts, and "wh-constructions" such as relative clauses.

a. Fragments

Approximately half of the sentences in the CASREPs are not full sentences. Nevertheless, these fragments follow quite regular patterns, and fall into one or another of four basic types: *tvo* (tensed sentence missing subject, as in A4.1.2, "Believe the coupling from diesel to sac lube oil pump to be sheared"); *zerocopula* (missing verb "be", as in A6.0.0, "Part ordered"); *nstg_fragment* (isolated noun phrase, as in B34.1.1, "Loss of oil pump pressure"); or *predicate* (isolated complement of verb "be", as in B12.1.2, "Believed due to worn bushings", or A.1.1.2, "Unable to consistently start nr 1b gas turbine").

The syntax and the semantics of these elements are quite regular, and thus fragment coverage does not add significantly to the complexity of the grammar. A total of six BNF rules (out of 106 total) and 3 restrictions (out of 55 total) were added to the grammar to cover fragments; in addition, 2 BNF rules and 1 restriction were altered to accomodate fragments.

b. Object options

The grammar has also been extended to cover a wider range of object types, including a variety of embedded infinitivals, embedded clauses, and non-clausal predications such as *subject+object of be* (as in B26.1.5, "High lo temp due to design of first flight oil cooler believed contributor to unit failure").

c. Sentence adjuncts

A rich variety of sentence adjuncts occur in the CASREPS, including a range of clausal and sub-clausal strings introduced by subordinating conjunctions (as in B20.1.1, "while engaged") and present participles (as in B11.1.1, "causing erratic operation"). In addition, the restriction component was developed to prevent spurious ambiguities arising out of the enrichment of sentence adjunct possibilities.

d. Wh-expressions

Although relative clauses and other wh-expressions are rare in the CASREPs (cf. B36.1.3, "65 psi which is low lube oil alarm set point"),the grammar has also been expanded to cover these constructions and to enforce the complex restrictions on their occurence.

3. PROBLEMS

The major remaining difficulties include the following:

a. Lexical scanner problems

Word-internal occurences of periods,slashes, etc. are currently rejected by the lexical scanner.

b. Xor problems

The 'committed or' which controls disjunctive application of the assertion, question, fragment, and compound options is generally successful in capturing the intended parse. However, there are several sentences in the CASREP corpus in which a spurious assertion parse preempts a correct fragment parse, e.g., B26.1.5, "High lo temp believed contributor to unit failure.", where "believe" is taken as the main verb with subject "temp" and "contributor" as the object ("they believed it"), rather than as a fragment of the type *zerocopula*, where "believed" is taken as a past participle ("temp [was] believed [to be] a contributor...").

c. Remaining grammar problems

Full and accurate coverage of the CASREPs requires further work on the grammar, including the following: finer-grained treatment of the noun phrase; restrictions on adverbs to prevent, e.g., the analysis of "very" as a sentence adverb; modification of the BNF rules to accomodate multiple sentence adjuncts; modification of conjunction rules.

CASREPS.TESTA
Summary of Parses
April, 1986

Sentences not preceded by casreps number are modifications of the original text. The rank of the correct parse is given in "Correct parse #" column. Note that these data reflect the grammar prior to the removal of xor from the fragment rule; therefore the figures for fragments do not include fragment parses subsequent to the correct one.

No.	Text	No.Parses	Times	Correct parse #
1.1.1	Starting air regulating valve failed.	5	1,3,6,8,10 (13)	4
1.1.2	Unable to consistently start nr 1b gas turbine.	1	2 (9)	1
1.1.3	Valve parts excessively corroded.	1	1 (2)	(N/G xor) 1
4.0.0	Tech assist requested.	1	2 (3)	1
4.1.1	While diesel was operating with sac disengaged, the sac lo alarm sounded.	1	10 (16)	1
4.1.2	Believe the coupling from diesel to sac lube oil pump to be sheared.	12	4,13,20,27,30,33, 37,43,49,52,56, 63,67 (69)	4
4.1.3	Pump will not turn when engine jacks over.	2	2, 4 (6)	1
5.0.0	Tech assist requested.	1	2(3)	1
5.1.1	Unable to maintain l.o. pressure to sac.	0		N/G scan
	Unable to maintain lo pressure to sac.	2	2,3 (6)	1
5.1.2	Disengaged immediately after alarm.	2	1,2 (2)	1
5.1.3	Metal particles in oil sample and strainer.	4	8,9,11,11 (15)	4
6.0.0	Part ordered.	1	2 (2)	1
6.1.1	Unable to maintain lube oil pressure to starting air compressor.	4	2,5,9,11 (36)	3
6.1.2	Inspection of lo filter revealed metal particles.	1	1 (4)	1
6.1.3	Retained oil sample and filter element for future analysis.	6	6,7,8,8,10,11 (13)	5
9.0.0a	Part fail.	1	2 (2)	1
9.0.0b	Part ordered.	1	1 (2)	1
9.1.1	Sac received high usage during two becce periods.	4	3,4,6,6 (7)	4
9.1.2	Ces received a report that lo pressure was dropping.	2	3,5 (7)	1
9.1.3	Alarm sounded.	1	1 (1)	1

No.	Text	No.Parses	Times	Correct parse #
9.1.4	Loud noises were coming from the drive shaft during coast down.	4	9,10,12,13 (17)	4
9.1.5	Drive shaft was found to rotate freely at the ssdg end.	2	2,5 (7)	2
9.1.6	Splines were extensively worn.	1	1 (1)	1
21.0.0	Assist required.	1	1 (2)	1
21.1.1A	Nr 4 sac oil pressure dropped below alarm point of 65 psig during monitoring of 1A gth.	6	11,18,23,39,43,48 (85)	2
21.1.1B	Start air pressure dropped below 30 psig during monitoring of 1A gth.	5	7,9,15,18,21 (42)	2
21.1.2	Oil is discolored and contaminated with metal.	3	1,3,3 (5)	3
22.0.0	Tech assist requested.	1	1 (2)	1
22.1.1	Loss of lube oil pressure during operation.	3	7,8,9 (12)	1
22.1.2	Investigation revealed adequate lube oil saturated with both metallic and non-metallic particles.	0		N/G scan
	Investigation revealed adequate lube oil saturated with both metallic and non-metallic particles.	10	23,24,*25,*25,*27, 29,30,*31,*32,*33, 38,39,40,42,43,44, 45 (54)	1
22.1.3	Request replacement of sac.	1	2 (4)	1
23.0.0	Assistance required.	1	2 (2)	1
23.1.1	The low lube oil pressure alarm and compressor fail to engage the alarm activated during routine start of start air compressor.	0		N/G input
	The low lube oil pressure alarm and compressor fail to engage alarm activated during routine start of start air compressor.	27	4,8,12,25,29,.... (215)	1
23.1.2	Metallic material was discovered in lo sump and filter assembly.	1	3 (11)	1
24.0.0	Require replacement.	1	1 (1)	1
24.1.1	Loss of lube oil pressure when start air compressor engaged for operation is due to wiped bearing.	4	4,5,23,26 (29)	N/G gram

No.	Text	No.Parses	Times	Correct parse #
24.1.2	Material clogging strainers.	2	3,4 (4)	1
25.0.0	Tech assist required.	1	1 (2)	1
25.1.1	During routine start of main gas propulsion turbine, sac air pressure decreased rapidly to 5.74 psi resulting in an aborted engine start.	0		N/G scan
	During routine start of main gas propulsion turbine, sac air pressure decreased rapidly to 5.74 psi resulting in an aborted engine start.	21	67,69,71...109 (227)	14
25.1.2	Exact cause of failure unknown.	1	2 (4)	1
25.1.3	Suspect faulty high speed rotating assembly.	1	2	1
28.0.0	Return to company.	3	2,2,3 (3)	1
28.1.1	Unit has excessive wear on inlet impellor assembly and shows high usage of oil.	2	5,12 (24)	1
28.1.2	Blades are bent and 1/4 inch deep chips are visible on leading edge.	2	2,3 (5)	1
30.0.0	Tech assist requested.	1	1 (2)	1
31.1.1A	Loss of second sac of two installed sac's.	2	2,4 (10)	N/G xor
31.1.1B	Unit has low output air pressure, resulting in slow gas turbine starts.	4	16,17,22,24 (46)	2
31.1.2	Troubleshooting revealed normal sac lube oil pressure and temperature.	3	4,4,5 (6)	1
31.1.3	Erosion of impeller blade tip is evident.	1	1 (3)	1
31.1.4	Compressor wheel inducer leading edge broken.	2	4,4 (6)	1

[1.1.1]
Note that only an adjectival reading is available for the prenominal analysis of "regulating".

[1.1.3]
Xor problem . Due to the optional intransitivity of "corrode", xor eliminates the correct zerocopula reading. However, this reading is close enough to qualify as correct.

[4.1.1]
Note that restriction {d_nullLnsr} removes rare gerund reading; {w_ving_lnr} thwarts an obscure analysis of ving as nvar.

[4.1.2]
The object is analyzable as nstgo, ntovo, or sobjbc. The latter possibility adds eight parses, but the object option sobjbe cannot be eliminated given, e.g., Testb 26.1.5 ("High LO temp .. believed contributor to unit failure").

[4.1.3]
"Over" is parsed first as an adverb preceding null object of "jacks". The most correct reading seems to be the second one, in which it is parsed as a particle; however, the sa reading is close enough to be counted as correct. If expressions such as "over" are reclassified as particles but not adverbs, in order to circumvent this, then they will have to be subcategorized for individually in the lexicon, which would lead to many false rejections of acceptable sentences.

[5.1.1]
Pu "to sac" is attached to rn in first parse (marked as correct here); but the second parse (with sa attachment of pn) seems more accurate.

[5.1.2]
In first parse, counted as correct, "immediately" is sa; perhaps the second, in which it is a left modifier of "after", is still more accurate.

[5.1.3]
I assume (without conviction) that npos "oil" should not be distributed over "strainer".

[6.1.1]
Although the third parse is listed as the correct one, the first parse is perhaps adequate: "to SAC" is attached to rn rather than sa.

[6.1.3]
The first parse differs from the correct one only in that it attaches "for future analysis" to rn rather than sa.

[9.0.0]
"Fail" is treated here as abbreviation for "failure". Or should these headers be treated as frozen expressions?

[9.1.1]
It is assumed here that the correct parse attaches "during NP" to sa and analyzes "two becce periods" as qpos + npos + nvar.

[9.1.2]
"That" is analyzable as determiner or complementizer.

[9.1.4]
"Coast down" is treated as idiom.

I assume that the most accurate parse (the fourth, counted as the correct one), attaches "from the drive shaft" to object, and "during coast down" to sa. However, the first parse might be sufficiently close, given the state of the system; it attaches the two pns to sa and rn, respectively.

[9.1.5]
Ambiguity: analysis of infinitive as sa (tovo) or passobj (correct).

[21.1.1A,B]
In the second se, counted as correct, "below"-phrase is sa rather than object (fifth parse).

[21.1.2]
The third parse is counted as correct, but the second parse, in which "with metal" is in sa, seems adequate.

[22.1.1]
The contextually correct nstg_frag parse is generated last; However, the zerocopula parse seems adequate, and is counted correct.

[22.1.2]
Conjunction . There are some analyses of "metallic" as avar preceding nulln that seem incorrect. This should be explored.

Object type . The nstgo object analysis seems somewhat more accurate than sven analysis here; within the venpass, the most accurate parse is perhaps the one in which "with ... particles" is attached as passobj rather than as sa. But the first parse, with sa attachment of this phrase, seems adequate.

Scanner problem . The problem remains that words containing "-" and such characters fail lookup because they are not atoms.

Conjunction . In order to parse the conjoined apos, larl has been defined as an lxr node. This may present a problem, since larl lacks a right adjunct.

Six other readings generated for this sentence contain conjoined lnr with nulln head of first lnr. Perhaps nulln should be disallowed in conjuncts unless it occurs in both: "There were five *(cats, and two dogs in the park"; "old and young were present", but *"old men and young were present" is quaint at best.

[23.1.1]
Input error . It is assumed that "the" preceding "alarm" is an error.

Re corrected version: The first six parses analyze "fail to engage" as an idiom (noun). In the remaining parses, "fail" is analyzed, legitimately, as the main verb (seven parses of conjoined subject x three parses of post-verb material).

[23.1.2]

Conjunction problem . Although the correct parse is generated, there is a missing parse, with conjoined npos "sump and filter". But since it seems unadvisable to allow full lnr in nnn, it's not clear how to modify the conjunction rules to allow for this reading.

[24.1.1]

Multiple rn : In the contextually correct reading, "loss" is modified by "of lube oil pressure" and the "when"-clause. However, multiple rn's are not permitted, except in the case of pn's. A semantically close reading in which the "when"-clause is an sa is also prevented, by {wmed_sa}, which rules out such sa's between subject and verb unless set off by commas (accounting for the ill-formedness of *"Louise when I called was tired"). The closest available reading actually generated is the second one, in which the when-clause is in the rn of "pressure".

Embedded fragment: "When sac engaged" seems most accurately parsed as an sven following "when". But in standard English, "when" cannot introduce an sven ("*I left when the car repaired"). Thus it may be that this corpus requires further modifications of the bnf rules beyond simply allowing matrix fragments. However, the optional intransitivity of "engaged" allows the material following "when" to be parsed as an assertion rather than an sven.

[24.1.2]

Perhaps an nstg_frag reading would be more accurate, but the first parse (zerocopula with objectbe-->vingo) seems close enough to be counted as correct. The second parse (zerocopula with objectbe-->nstg) seems more questionable; perhaps {w_nonnull_ln} should be strengthened to require material in qpos or tpos rather than simply ln. (This decision depends on judgments about acceptability of, e.g., "Sen.Jones complete idiot").

[25.1.1]

Scanner problem . The decimal point cannot currently be entered.

The long time to first parse may reflect the fact that the sentence is an extensive garden path, since the main verb "decreased" may initially be mis-analyzed as a participle in rn.

The parses generated prior to the correct fourteenth parse analyze the nvar of the subject as either "resulting" or nulln rather than "psi".

[28.0.0]

Third parse is questionable: objbe in zerocopula (analogous to "house in an uproar", or "trip to Texas, not Arizona").

[28.1.1]

First parse (counted as correct) attaches "on...assembly" to rn; sa attachment, as in second parse, might be considered the more accurate parse.

[31.1.1A]

Lexical entry procedure should be modified to generate "'s" plurals routinely for abbreviations.

Xor problem . The contextually incorrect assertion parse preempts the nstg_frag parse that is intended here.

[31.1.1B]

Here the attachment of the pn in object or sa seems important, as "result in" has an

idiomatic meaning. Thus the first parse, with sa attachment, is not counted as correct.

[31.1.4]
"Leading edge" is entered in the lexicon as an idiom, as a result of its occurence here in nvar position. ("Leading" could only be parsed as avar, an impossibility here given that it follows a series of npos elements.) Occurence in compounds seems a potential test for fixed phrases; compare this sentence with the less acceptable "*peach poisonous pits are dangerous" (vs. "peach pits are dangerous").

CASREPS.TESTB
Summary of Parses
April, 1986

Sentences not preceded by a casreps number are modifications of the original text. The rank of the correct parse is given in "Correct parse #" column. Note that these data reflect the grammar prior to the removal of xor from the fragment rule; therefore the figures for fragments do not include fragment parses subsequent to the correct one.

No.	Text	No.Parses	Times	Correct parse #
2.0.0	Replacement requested.	1	1 (2)	1
2.1.1	Loss of lube oil pressure during operation nr. 2 ssdg.	8	13,19,19,22,25, 26,30 (34)	N/G input(+scan)
2.1.2	Metal particles found in lube oil filter.	1	12	1
3.1.1	Gas turbine starting air compressor inoperative.	2		1
3.1.2	Power pack failed.	1	1 (1)	1
7.0.0	Assistance requested.	1	1 (2)	1
7.1.1	Sac had local monitoring capacity for lube oil pressure only, due to the recent failure of the sac lube oil pressure transducer.	1	10 (50)	1
7.1.2	Prior to engagement it was reported that sac lo pressure dropped to zero.	2	2,3 (8)	1
7.1.3	No metallic particles in oil filters.	2	*4,5 (6)	2
7.1.4	Borescope investigation revealed a broken tooth on the hub ring gear.	4	5,7,8,10 (11)	1
7.1.5	It is likely the lo pump has sheared.	1	2 (4)	1
7.1.6	The lo pressure and alarm capability is a necessity for operation.	4	1,2,3,4 (6)	N/G input
7.1.7	Drive shaft for sac was manufactured locally.	1	1 (3)	1
7.1.8	S/F reinstalled old sac utilizing new drive shaft.	0		N/G scan
	Fe reinstalled old sac utilizing new drive shaft.	3	3,5,6 (7)	3
7.1.9	On testing of sac lube oil pressure could not be adjusted above 35 psig.	3	2,6,9 (18)	3
7.1.10	Replacement sac will be required.	1	1 (2)	1

No.	Text	No.Parses	Times	Correct parse #
7.1.11	The original drive shaft, when installed, was packed utilizing 60 grams of grease, when removed, on failure of sac, the drive shaft was dry and showed signs of extensive heat stress.	?	1460 ...	3
	The original drive shaft, when installed, was packed utilizing 60 grams of grease.	3	3,*4,*5 (7)	1
	When removed, on failure of sac, the drive shaft was dry and showed signs of extensive heat stress.	2	11,15 (25)	1
8.0.0	Tech assist requested.	1	2 (2)	1
8.1.1	Loss of one of two starting air compressors.	?	12, ...	1
8.1.2	Low speed coupling from diesel to sac lube oil pump failed.	6	*2,*9,14,21,26, 33 (36)	4
10.0.0	Tech assist requested.	1	2 (2)	1
10.1.1	HBV failed, causing spline assy to fail causing damage to the sac.	4	3,4,7,8	1
11.0.0	Tech assist required.	1	2(2)	1
11.1.1	Compressor will not remain fully engaged causing erratic operation, surging, and a hazard to personnel and equipment.	55	8,9,10,11....115 (126)	4
12.0.0	Tech review required.	1	2 (2)	1
12.1.1	Sac lo pressure decreases below alarm point approx. seven minutes after engagement.	0		N/G scan.
	Sac lo pressure decreases below alarm point approx seven minutes after engagement.	4	3,7,13,13 (17)	2
12.1.2	Believed due to worn bushings.	4	2,2,4,5 (6)	2
13.0.0	Must be removed.	1	1 (1)	1
13.1.1	Loss of sac oil pressure dropped to 72 psi then increased to 90 psi and then failed while starting gas turbine.	0		N/G gram(+input)
	Loss of sac.	1	2 (3)	1
	Oil pressure dropped to 72 psi then increased to 90 psi and then failed while starting gas turbine.	21	38 ... 171 (187)	N/G
14.0.0	Req tech assist.	1	1 (1)	1
14.1.1	Loss of one of three start air compressors.	2	12,13	1

No.	Text	No.Parses	Times	Correct parse #
	Oil pressure has dropped to 72 psi then increased to 90 psi and then failed while starting gas turbine.	27	8 ... 175 (179)	1
14.1.2	Starting air compressor engaged for approx two minutes when lube oil pressure dropped below 65 psi alarm setting.	?	21,25,28,32,44,...(434)	5
14.1.3	Compressor could not be disengaged from either remote or local control location, for approx three minutes following low lube oil pressure alarm.	12	3,6,13,15,38,40,47, 49,71,73,80,82 (132)	1
14.1.4	Lube oil is very dark in appearance and has burnt odor.	4	*2,*4,7,9 (11)	3
15.0.0	Tech assist requested.	1	1 (2)	1
15.1.1	Reliability of third of three sac's suspect - if unit fails unable to start main propulsion gas turbines.	?	36 (58)	1
15.1.2	Color of 23699 oil indicates overheating of sac, oil pressure normal.	1	17 (20)	1
16.1.1	During normal start cycle of 1A gas turbine, approx 90 sec after clutch engagement, low lube oil and fail to engage alarm were received on the acc.	over 30	162 to 1st parse	N/G gram
16.1.2	All conditions were normal initially.	1	2 (2)	1
16.1.3	Sac was removed and metal chunks found in oil pan.	0		N/G gram
	Sac was removed and metal chunks were found in oil pan.	1	2 (5)	1
16.1.4	Lube oil pump was removed and was found to be seized.	2	3,4 (6)	1
16.1.5	Driven gear was sheared on pump shaft.	1	1 (3)	1
17.0.0	Tech evaluation req.	1	2 (2)	1
17.1.1	Loss of one of three sac's - routine visual inspection during normal engine operation revealed gear housing cracked.	2	47,56...	1

No.	Text	No.Parses	Times	Correct parse #
17.1.2	Engine secured, detailed inspection revealed large crack in gear housing on aft end and broken marmon clamp flange on surge valve outlet.	22?	2072, ...	11?
	Engine secured.	1	1 (1)	1
	Detailed inspection revealed large crack in gear housing on aft end and broken marmon clamp flange on surge valve outlet.	Over 22	215,216,...	11
18.0.0	Item canabilized.	0		N/G input
	Item cannibalized.	1	4 (5)	1
18.1.1	Cannibalized sac for use on USS Duncan.	4	14,17,22,24	4
19.0.0	Part ordered.	1	1 (2)	1
19.1.1	Redneed capability of nr 4 sac restricts ships operation.	0		N/G input
	Reduced capability of nr 4 sac restricts ship's operation.	1	4 (9)	1
19.1.2	Extended use of nr 4 sac has resulted in periodic low lube oil pressure alarm.	3	7,16,21 (26)	2
19.1.3	Lube oil change, filter change, and adjustment of pressure regulator have had no impact on lube oil pressure.	?		4?
19.1.4	Three minutes is the maximum time nr 4 sac can be operated in a non-alarm condition.	0		N/G scan
	Three minutes is the maximum time nr 4 sac can be operated in an alarm condition.	2	4,8 (14)	1
20.0.0	Tech assist req.	1	2 (2)	1
20.1.1	During gth motor start, air pressure dropped below 30 psi and oil pressure decreased slowly to 70 psi, while engaged.	Many	102 to 1st parse	4th+
	During gth motor start, oil pressure decreased slowly to 70 psi, while engaged.	1	11 (25)	1

No.	Text	No.Parses	Times	Correct parse #
20.1.2	Metal particles found in oil sample.	2	10,12 (17)	1
26.0.0	Technical assistance requested.	1	2(2)	1
26.1.1	Reduced capacity of one of three sac's.	4+	13,14,24 ...	3
26.1.2	Cannot engage sac for extended period of time due to increased lo temp and sharp decrease in lo pressure.	30+	4 ...	4th+
26.1.3	Metal contamination in lo filter.	2	*4,6 (8)	2
26.1.4	Internal part failure.	1	2 (2)	1
26.1.5	High lo temp due to design of first flight oil cooler believed contributor to unit failure.	4	4,6,44,46 (80)	N/G xor
27.0.0	Part ordered.	1	2 (2)	1
27.1.1	Experienced loss of sac lube oil pressure and self-disengagement immediately following clutch engage command.	0		N/G xor(+scan)
	Experienced loss of sac lube oil pressure and self disengagement immediately following clutch engage command.	4	26,28,32,33 (48)	N/G (xor)
27.1.2	Sac apparently seized during clutch engagement causing input drive shaft to remain stationary while drive adapted hub on ssdg continued to rotate.	0		N/G input
	Sac apparently seized during clutch engagement causing input drive shaft to remain stationary while drive adapter hub on ssdg continued to rotate.	8	4,6,18,20,50,51, 59,60 (133)	6
27.1.3	Drive shaft sheared all internal gear teeth from drive adapter hub.	3	3,5,7 (9)	1
29.0.0	Technical assist requested.	1	2 (2)	1
29.1.1	Fct open and inspect, revealed bearing material on bottom of strainer.	0		N/G input
	Fct open and inspect revealed bearing material on bottom of strainer.	2	4,6 (12)	1
29.1.2	After flushing unit, engaged pressure dropped to 62 psig within 45 seconds of engaging sac.	16	168,171,180,181,184, 186,192,194,284,286, 298,301,304,309 (314)	8
29.1.3	Disengaged pressure satisfactory.	1	2 (3)	1
30.0.0	Technical assistance requested.	1	2 (2)	1

No.	Text	No.Parses	Times	Correct parse #
30.1.1	Loss of one of two sac's.	2	10,11 (14)	1
30.1.2	Unit has low output air pressure, resulting in slow gas turbine starts.	4	18,20,26,28 (52)	2
30.1.3	T/S revealed normal sac lube oil pressure/temperature.	0		N/G scan
	Troubleshooting revealed normal oil pressure.	1	2 (3)	1
30.1.4	Impellor blade tip erosion evident.	1	3 (5)	1
30.1.5	Sac beyond shipyard repair.	1	5 (5)	1
30.1.6	Cause of erosion of impellor blades, undetermined.	1	3 (8)	1
30.1.7	Second generation sac received onboard for installation.	5	5,7,10,12,14 (15)	2
32.1.1	Loss of 50 percent of start air capability.	1	13...	1
32.1.2	Nr 2 sac can be operated at reduced capacity.	1	2 (3)	1
32.1.3	This situation present potential over temp hazard to lm2500 during start up evolutions and further degradation of mobility.	?		N/G input
	This situation presents potential over temp hazard to lm2500 during start up evolutions and further degradation of mobility.	over 90	2...	12?
32.1.4	Difficulty began with audible pulsations in compressor outlet air pressure under steady state conditions.	8	12,15,19,23,25,28,30,32 (34)	6?
32.1.5	Cause of casualty unknown.	1	2 (3)	1
33.0.0	Request shipyard replace.	1	1 92)	1
33.1.1	Oil pressure has been slowly decreasing.	1	1 (3)	1
33.1.2	Failure occurred during engine start when oil pressure dropped below 60 psig.	4	3,4,8,9 (12)	4
33.1.3	Investigation revealed excessive fine metal particles in oil.	2	3, 4 (5)	1
34.0.0	Assistance requested.	1	1 (2)	1
34.1.1	Loss of oil pump pressure.	?	8 ...	1
34.1.2	Suspect sheared connecting pin in pump drive assembly.	2	10,11 (17)	1
34.1.3	Loss of pressure was sudden and unexpected.	1	1 (2)	1

No.	Text	No.Parses	Times	Correct parse //
34.1.4	Investigation by todd revealed sac spline input drive shaft disconnected from diesel hub.	4	11,13,15,17 (25)	4
34.1.5	Hub assembly and spline shaft erroded beyond use.	0		N/G input
34.1.5	Hub assembly and spline shaft eroded beyond use.	2	4,*8 (10)	1
34.1.6	Todd LA to replace worn hub assembly and spline shaft.	7	15,18,19,26,27,31, 31 (34)	5
35.0.0	Parts ordered.	1	1 (2)	1
35.1.1	Experienced total loss of sac lo pressure and self disengagement while conducting gte water wash.	8	55,60,67,71,84,89,94,98 (105)	2
35.1.2	Investigation revealed stripped lo pump drive gear and hub ring gear.	2	7,8 (13)	1
36.0.0	Tech assist.	1	1 (1)	1
36.1.1	A number of slow gas turbine starts has been noted recently using 13 sac.	2	4,5 (9)	1
36.1.2	A trend of increasing lube oil temperature and decreasing lube oil pressure dictated cleaning the lube oil cooler and replacing the lube oil filter as corrective maintenance.	?	212, ...	4
	A trend of increasing lube oil temperature and decreasing lube oil pressure dictated ... replacing the lube oil filter as corrective maintenance.	over 30	26...	?
	A trend of increasing lube oil temperature ... dictated cleaning the lube oil cooler ... as corrective maintenance.	8	10,*12,*13,15,46, 48,50,51 (90)	4
36.1.3	After the maintenance was accomplished, operational tests revealed low lube oil pressure (65 psi which is low lube oil alarm set point) before the required three minute sac engaged time limit had run out.	19	114,125,131,140...	4
36.1.4	The lube oil filter was opened up revealing minute metallic particles.	4	2,5,8,12 (16)	4
36.1.5	Indications are that a new lube oil pump is required.	1	3(4)	1
36.1.6	Guarantee deficiency.	1	1 (1)	1

[2.1.1]

Scanner problem. Period in abbreviation prevents parsing.

Structure of NP . In the closest parse obtained (the second), "operation nr. 2 ssdg" is parsed inaccurately with "operation" in npos modifying the namestg. However, introduction of implicit "of" seems ill-advised as a means of coping with this non-standard input.

[2.1.2]

Adverb problem . Restriction {d_d_or_p} prevents analysis of "in" as adverb.

[7.1.1]

"Only" is parsed somewhat questionably as an adjective in rn. "Monitoring" can only be parsed prenominally as adjective, not nvar.

[7.1.3]

One might argue that the second (nstg_frag) parse (with sa attachment of the prepositional phrase) is more accurate than the first parse (in which it is attached to rn), but the first is counted as correct.

[7.1.4]

Again, one might argue that the second parse (with the prepositional phrase in sa) is more accurate than the first parse (in which it is attached to rn), but the first is counted as correct.

Note that the ambiguity of "broken" as *ven or *adj doubles the parse count.

[7.1.6]

Number agreement. The grammatical error in this sentence is not the cause of its unparsability. (Note that {wagree} has had to be relaxed at least for "be", given grammatical sentences such as "ten minutes is the limit". In fact, not only "be" allows plural subjects with singular verb; cf. "ten minutes of listening to his chatter really taxes me to the limit". It seems to be a function of the semantics of the subject rather than the verb.) Thus the error in this sentence does not present it from being parsed.

The sentence as it stands seems incoherent. If it is taken as "[the (correct) lo pressure] and [alarm capability]", i.e., with an implicit modifier "correct", then the correct parse is the first one. And clearly it is unlikely that the correct reading is the one paraphraseable as "the capacity for lo pressure and alarm". Another possibility, suggested by NYU, is that "and" is a typographical error.

[7.1.7]

"Sitrep 002:" is not treated as part of the sentence proper.

[7.1.8]

Scanner problem. "/" cannot be input.

[7.1.8]

"Utilizing" could be legitimately analyzed as noun modifier in apos or rn, or (correctly here) as sentence adjunct.

[7.1.9]

I assume that on the correct parse "lube oil pressure" is the subject. The second and third parses divide up the string of nouns differently between sa and subject.

[7.1.11]

In fact, this parses as a compound; correct parse is 3rd. Time: 1,460 sec!

Punctuation error is assumed for 7.1.11. Thus the comma preceding "when" has been changed to a period, as indicated, and 7.1.11 has been broken into two clauses to test its parsability in the absence of this error.

Second clause : The second parse for this clause is the correct but contextually incorrect analysis of the object as nn rather than nstgo.

[8.1.2]

Adverb problem. "Low" is mis-analyzed as adverbial sa in first two parses.

[11.1.1]

The first three parses are correct but distribute ln incorrectly ("surging" should be local, I assume).

The massive number of parses appears to be a function of conjunction; whether there are, in fact, 55 distinct and grammatical analyses remains to be determined. In the absence of the conjoined material (that is, with the first comma and everything to its right deleted), there are only three parses.

[12.1.1]

Scanner problem. "." cannot be input.

The second two parses take "point" as the (arguably intransitive) main verb.

[13.1.1]

Punctuation error. This sentence is ungrammatical as punctuated. It has been reanalyzed into two clauses. However, it may still be unacceptable: "failed" would seem more likely to take the sac, rather than the oil pressure, as its subject.

Second clause : Conjunction problems . For some reason, three assertions are not parseable in conjunction rules generated from this grammar. This forces 13.1.1b to be parsed as three ltvr's, but the absence of rv prevents the attachment of their pn's ("to 72psi", etc). Thus only the readings in which "increased" and "decreased" are past participles in rn remain. With the addition of "has" (see table), the correct parse is, in fact, the first parse generated.

Also, "then" (but not "and") as the conjunction allows for an incorrect reading in which a "copied nullobj" is created in the first conjunct.

[14.1.2]

Perhaps the sixth (rather than the fifth) parse is the most accurate, since it attaches "below" as object rather than sa. In general, sa attachment of subcategorized-for pn's is not regarded as an error, unless the verb + pn form a virtual idiom.

The variety of parses arises from the different attachment possibilities of "for two minutes", the "when"-clause, "below 65 psi"; the two analyses of "65 psi alarm setting"; and the analysis of subject as gerund or lnr.

Also, the fact that the entire sentence can be initially misanalyzed as an lnr contributes to the long parsing times.

[14.1.3]

Two ambiguities in the absence of conjunction (nominal or adjectival analysis of "low", "following" as *p or *ving) combine with a three-way conjunction ambiguity ("remote or local" analyzed as a conjoined adjadj or a conjoined lnr, the first one headed by nulln). With the latter, there is the ambiguity between distributed or local scope for tpos,qpos. The correct parse is assumed to be the first, in which "low" is adjectival,"following" is a preposition, and "remote or local" is a conjoined phrase in adjadj.

[14.1.4]

In the parse listed as correct, "in appearance" is a sentence adjunct. However, the fourth parse, in which it is a right adjunct of the adjective "dark", is probably still more accurate.

Adverb problem . Clearly a finer-grained analysis of adverbs is necessary. In the first two parses. "very" is analyzed incorrectly as an adverb. The adverb features developed by Sager will clearly prove useful here, but there are difficulties in applying them. There is no one feature which is associated with all and only those adverbs which are acceptable in sa position. For example, not all adverbs which may occur in sa position are marked with the feature "dsa": neither "yet" (as in "She has not eaten lunch YET") and "there" (as in "He was happy THERE") is dsa. There is a group of features one or another of which characterizes any adverb which may appear in sa; this group includes dsa. dlv, drv, drw. However, any adverb input by the SDC lexical entry procedure has an empty feature list, so a restriction limiting adverbs in sa to those bearing one of these features would require considerable lexical work. Finally, an attempt to exclude sa analyses of adverbs like "very" by forbidding adverbs with certain features (such as dla -- left adjunct of adjective) will prove too strong, since, e.g., "always" is marked with the feature dla as well as drv/drw/dlv.

[15.1.2]

I assume that "oil pressure normal" is not to be taken as part of a conjoined object of "indicate", as the color of oil would not be an indicator of oil pressure. Thus the analysis of this sentence as a compound is assumed to be correct.

First clause : Shapes needs to be developed so as to recognize part numbers for this domain. Currently, 23699 is parsed only as qpos.

[16.1.1]

This sentence presents a number of difficulties.

Grammatical error . "Alarm were received" should perhaps be "alarms were received",

Multiple sa's . The correct analysis of this sentence would seem to involve two initial sa's, something currently disallowed by the grammar. Thus "approx 90 sec after clutch engagement" is incorrectly parsed as an appos attached to "turbines".

Treatment of apposatives : This points up the inadequacy of the current appos rule, which substitutes for rn and is therefore not associable with a head noun which itself contains an rn.

Conjunction. The rules do not currently allow for conjoined ln, so that "(low lube oil) and (fail to engage) alarms" cannot be correctly parsed. (And the contextually appropriate parse of "disk and sac alarms are required" cannot be generated.)

There are an extraordinarily large number of parses in which the conjunction is associated with the introductory pn in sa, the subject being "alarms".

Structure of NP . Also, the bnf rules do not currently allow for modification in npos, as in "low lube oil alarm".

This sentence clearly requires further work, because of the indequacy of the parses obtained and the very long parsing times.

[16.1.3]

Conjunction. This sentence does not parse without addition of "were" to second conjunct. Conjunction rules do not seem to handle (verb) gapping, even without the "sloppy identity" that holds here between the overt and implicit instances of "be". ("Sac was repaired and disk replaced" is also rejected.) We could allow "and" to join conjuncts, but this seems dubious: cf. "sac was repaired - replacement of blade" vs *"sac was repaired and replacement of blade".

[17.1.1]

This is parsed as a compound, with nstg_frag the first element. The second parse is the more accurate one: "one" is in qpos modifying nulln. (In the first parse, "one" is the head nvar.) As with other fragments, only parses with the first fragment option to succeed are listed in table.

Note that zerocopula reading of first conjunct is ruled out by assorted heuristics ({d_of}, {w_nonnull_ln}).

[19.1.1]

Punctuation. Apostrophe must be added.

[19.1.3]

Appos . The first three readings construe the second conjunct as an apposative on the first; appos and null options in rn should probably be re-ordered.

Conjunction . Are the twelve conjunction readings distinct possibilities? The contextually correct reading comes late because earlier readings copy the pnpn attached to the final conjunct ("of pressure regulator") into earlier conjuncts, while the correct reading would seem to be the local one.

[19.1.4]

Scanner problem. Word-internal dash not currently recognizable.

Wagree . Sentences such as this require that wagree be relaxed to allow plural subjects with "is". (Cf. "Peanut butter and pickles is a horrible combination").

[20.1.1]

Conjunction . Parsing times seem extraordinarily long for this sentence, even given its numerous unexpected conjunction ambiguities (the initial pn may be taken as containing three conjoined NPs; the first four readings, for example, take "start" as the first of three conjoined NPs; the next five or more take "pressure" as subject).

Upon removal of the first conjunct ("air pressure dropped below 30 psi"), a single (correct) parse is generated in 11 sec (25 to NMP), as indicated in table.

[26.1.2]

The conjunction and pn attachment possibilities in this sentence are legion, and have not all been examined; in addition, there is an ambiguity between npn (contextually inappropriate) and nstgo object analyses. (The npn object option has pval "in", as in "We engaged them in conversation".)

[26.1.5]

Xor problem . Because there is an assertion reading (with "contributor" the nstgo object of active "believed"), the correct zerocopula parse (in which "contributor" is the remanants of active sobjbe) is not generated. However, selection can easily rule out this reading.

[27.1.1]

Xor problem. Because there is an assertion parse (with "engage" as main v..b), the contextually correct two parse is not generated.

Re the long time to first parse: note that the analysis of "experienced" as prenominal *ven creates a severe garden path.

[29.1.1]

Input. Punctuation error.

"Open and inspect" entered as idiom in lexicon.

[29.1.2]

The extraordinarily long parsing time for this sentence needs to be investigated. (Note that it does present a considerable garden path to the parser, since the entire string "engaged .... sac" could be analyzed as an NP.)

The various analyses depend upon analysis of the two [ving nvar] sequences as lnr or gerund (in both cases) and on pn attachment. The selection of the eighth parse as the correct one needs to be verified (accidental logout prevented closer inspection).

[30.1.1]

Structure of NP . "One" can be parsed as nvar (in first parse) or q. I mark the first parse as correct, though presumably the second is the truly correct one. Will this create dufficulties for semantics?

Note that zerocopula analysis is prevented by requirement that predicate nominal have nonnull ln (compare "party a disaster" with "party disaster").

[30.1.3]

Scanner problem. Word-internal slashes not accepted.

[30.1.4]

Note that the requirement that ln be nonnull, {w_nonnull_ln}, eliminates other zerocopula analyses.

[30.1.5]

{w_nonnull_ln} eliminates other zerocopula readings.

[30.1.6]

Comma is now allowed post-subject in zerocopula, which may add considerably to the number of parses for zerocopulas and compounds.

[32.1.3]

Grammatical error . "Present" should have been "presents". I assume (without conviction) that "over temp" is equivalent to "overheating"; thus it is entered in the lexicon as an idiom.

The nn subcategorization for "present" has been removed from the lexicon, as it sounds ungrammatical to me and contributes an additional 20 parses to this sentence. However, addition of npn subcategorization adds parses.

The variety of parses arises from the various pn attachment possibilities, the ambiguity of "start up" as an idiom or noun followed by preposition; and, of course, the scope possibilities associated with conjunction.

[32.1.4]

The meaning of this sentence is unclear: are the pulsations really pulsations in *air pressure* ? The sentence as punctuated would seem to have no other analysis.

The correct analysis is assumed (without conviction) to be that in which "with" is in pn object of "begin" and "under" is in sa.

[33.0.0]

"Replace" would have to be entered as a noun to parse this header, but see 34.1.6 for consequences of this.

[33.1.1]

The gerund and ving/nvar readings are prevented by {d_nullLnsr} and {w_ving_lnr}.

[33.1.2]

Although the fourth parse is listed as the correct one ("when" in sa, "below" in object), the first parse might be adequate ("when" in rn, "below" in sa).

[34.1.4]

Although the fourth parse (sven object, "from" in pn object) is listed as the correct one, the first is perhaps adequate: nstgo object, "from" in sa.

[34.1.6]

What is "LA" here? Part of "TODD"? An abbreviated predicate of some sort? A locative phrase? It is treated here as simply *n.

The first four analyses can be eliminated if "replace" is not categorized as a noun (necessary for 33.0.0, which is perhaps a frozen p' ase anyway; perhaps an elliptical tv).

[36.1.1]

Wagree should perhaps be modified to allow for plural verbs following phrases like "a number of NP". (In this case, however, the verb is singular.)

Shapes (?): "13 sac" is parsed incorrectly as [qpos + nvar]; a more complete treatment of equipment names in this corpus is in order.

{*d_lv*} should be modified to rule out second parse in which "recently" is in lv of "using".

[36.1.2]

Note very long time (219 sec) to first parse. Correct parse is fourth.

{d_init_sa} disallows the reading(s) in which conjoined lnr's are flanked by vingo sa's.

"As" is (incorrectly) treated as a conjunction in certain parses because it is listed in the lexicon as a spword.

[36.1.3]

Adverb problem . Again, "low" is parsed as an adverb in sa in the first reading.

The very long parsing times need to be examined. (Note that times are shortened by adding "sac engaged time limit" as an idiom -- first parse in 71 sec, 100 sec to correct parse -- rather than parsing sac (oddly but not really inadequately) as an [lcda + ven].)

Also, there appear to be some duplicate parses.

[36.1.4]

The various well-formed but contextually incorrect parses generated include analysis of "up"

as preposition (rather than particle), and of "revealing ..." as a gerund. (cf. "For years they talked about revealing the secret of their great wealth")

[36.1.6]
What does this mean? Xor will only allow the tvo reading.